



ИПМ им.М.В.Келдыша РАН

Абрау-2017 • Труды конференции



Е.М. Лаврищева, Л.Е. Карпов,
А.Н. Томилин

**Подходы к представлению научных
знаний в Интернет науке**

Рекомендуемая форма библиографической ссылки

Лаврищева Е.М., Карпов Л.Е., Томилин А.Н. Подходы к представлению научных знаний в Интернет науке // Научный сервис в сети Интернет: труды XIX Всероссийской научной конференции (18-23 сентября 2017 г., г. Новороссийск). — М.: ИПМ им. М.В.Келдыша, 2017. — С. 310-326. — URL: <http://keldysh.ru/abrau/2017/68.pdf> doi:[10.20948/abrau-2017-68](https://doi.org/10.20948/abrau-2017-68)

Размещена также [презентация к докладу](#)

ПОДХОДЫ К ПРЕДСТАВЛЕНИЮ НАУЧНЫХ ЗНАНИЙ В ИНТЕРНЕТ НАУКЕ

Е.М. Лаврищева¹, Л.Е. Карпов², А.Н. Томилин³

¹Е.М. Лаврищева <lavr@ispras.ru>

*Институт системного программирования РАН,
Московский физико-технический институт (технический университет),*

²Л.Е. Карпов <mak@ispras.ru>

³А.Н. Томилин <tom11@bk.ru>

*Институт системного программирования РАН,
Московский государственный университет имени М. В. Ломоносова,*

Аннотация. Дано определение Интернет науки, получившей название e-science, и ее целей, приводится перечень базовых организаций, занимающихся ее развитием. Представлены основные научные понятия этой науки – информация, данные, знания. Определена схема научного процесса формирования знаний в e-science. Дан анализ состояния e-science и пути развития ресурсов семантической сети (Семантик Web), включая готовые ресурсы и сервисы. Дано определение базовых понятий этой науки и семантического Grid, включающего ресурсы, сервисы и онтологию управления знаниями. Приведена архитектура Web-Grid, содержащая набор программных, сервисных и интерфейсных ресурсов, хранящихся в репозитории, брокер сервисов и инструментов. Рассмотрены перспективные пути развития e-науки.

Ключевые слова: знание, представление, извлечение знаний, сервис, ресурс, программная система, Веб-система, Grid –система, брокер сервисов, протоколы Интернет.

Введение¹

Электронная наука (e-science) была определена директором Главного управления по науке и технологиям Великобритании Дж. Тейлором в связи с развертыванием программы глобализации науки в Соединенном королевстве в 2000 году. В это время для обозначения науки в других странах Европы использовались термины *e-исследования* (e-research), а в США *e-Инфраструктура* (e-Infrastructure) и *e-Киберинфраструктура* (e-Cyberinfrastructure) [1-4].

Потребность в подходах, получивших наименование «e-наука» Интернет, возникла в связи с массовым переходом к проведению глобальных вычислений в современных сетевых средах с использованием вычислительных ресурсов и многочисленных данных в реальном масштабе времени. Основу этой науки составляет технология глобальной информационной сети (Интернет), вычисления с использованием кластеров слабосвязанных гетерогенных вычислительных машин (Grid-вычисления) и глобальные сетевые средства доступа к мощностям миллионов машин мирового пространства для производства вычислений и хранения огромных массивов данных, полученных в результате их обработки. Эта наука ориентирована на проведение научных экспериментов и развитие таких отраслей науки, как:

- медицина и здравоохранение (разработка средств наблюдения, диагностики и лечения);
- биология и биоинформатика (исследование генома человека для понимания генетических болезней, создание новых лекарств и др.);

¹ Работа поддержана грантами Российского фонда фундаментальных исследований № 16-01-00352, № 15-07-02355.

- нанотехнологии (разработка новых материалов на молекулярном и атомном уровне);
- математическое моделирование в физике, метеорологии, экологии, геологии, астрономии, космосе;
- инфраструктуры киберсистем и др.

В составе электронной науки выделены такие направления, как информатика, системная и программная инженерия, компьютерная наука или программирование (computer science), семантическая сеть (Semantic Web), облачные вычисления (Cloud Computing), большие данные (Big Data), Data Mining, Ontology и многое другое.

Электронная наука ориентирована на проведение современных научных исследований, включая подготовку и проведение экспериментов, сбор данных, распространение результатов, а также их хранение и доступ ко всем полученным материалам. Исследования проводятся методом моделирования, анализа данных и информации для представления накопленных знаний в разных областях [1, 2]. Главными организациями развития Науки (e-science) в Интернет являются:

- Европейский центр по ядерным исследованиям ЦЕРН (European Organization for Nuclear Research, CERN), объединяющий более 120 стран-участниц, более 20 стран-наблюдателей, включая Россию, а также несколько международных организаций (ЮНЕСКО и др.);
- UK e-Science Programme (Великобритания) (<http://www.escience-grid.org.uk/>) и Core Programme (eSCP), объединяющие более 20 научных центров во главе с Национальным центром электронной науки в Эдинбурге (National e-Science Centre, NSC);
- общественные организации Глобальный форум Grid (GGF, Global Grid Forum), DataGrid и Globus Toolkit (<http://www.globus.org>);
- Консорциум Европейского союза DEISA (Distributed European Infrastructure for Supercomputing Applications);
- Проекты Tera-Grid, Strategic Grid Computing Initiative и Grid-сети комитета Национального управления по воздухоплаванию и исследованию космического пространства (National Aeronautics and Space Administration, NASA) и министерства обороны США;
- Китайский проект China-Grid;
- Индийский национальный grid-проект GARUDA;
- Проект создания единой компьютерной сети GLORIAD, включая США, Канаду, Европу, Россию, Китай, Южную Корею и др.

На конференции «Е-наука 2014» комитет IEEE дал новый вариант ее определения: «Электронная наука включает в себя то, что называют большие данные, большой адронный коллайдер ЦЕРН, который в год производит около 780 терабайт научных данных для интенсивной обработки в вычислительной биологии, биоинформатике, геномике и в социальных науках...» [4]. Главным аспектом электронной науки является научный эксперимент [3-22].

1. Научные эксперименты в электронной науке

В электронной науке для проведения научных экспериментов разработана общая схема (рис.1) [5, 6].

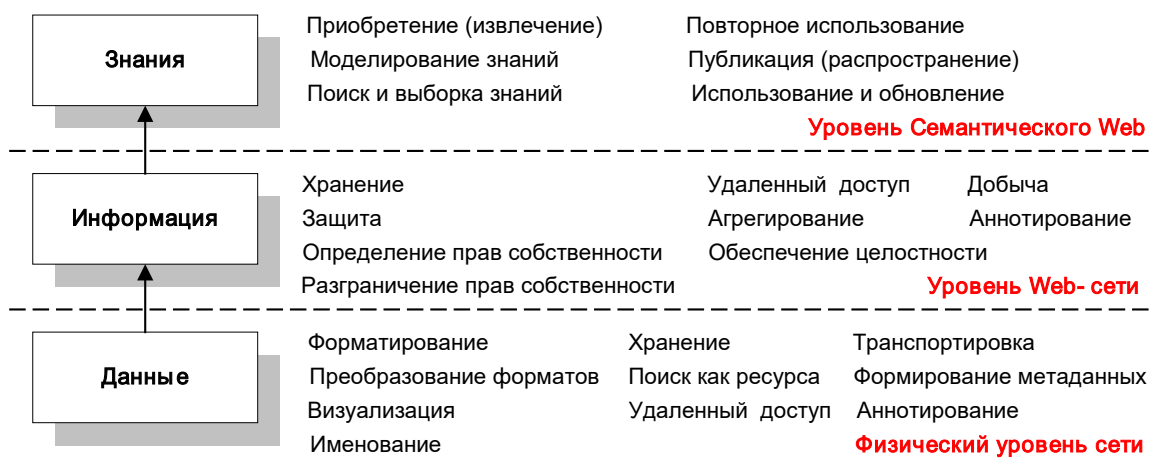


Рис. 1. Три процесса проведения научных экспериментов

Основные элементы этой схемы это:

- Знания, как прикладная информация, которая необходима для достижения цели решаемой научной проблемы или определения путей ее решения;
- Информация, как данные, которые несут в себе смысл (например, содержимое (контент) документа, метаописание данных, их значения и др.);
- данные, как совокупность битов и байтов (например, запись базы данных, документ в файле, мето-поле базы данных, файл изображения и др.), которые не интерпретируются.

Для знания (knowledge) предложен жизненный цикл (рис.2), включающий процессы:

1. *Приобретение (извлечение) знаний (knowledge acquisition)*. Этот процесс превращает имеющуюся информацию в конкретные знания, которые можно использовать для решения отдельных задач. При этом проводится переработка огромного объема данных и разной избыточной информации. Приобретение знаний заключается в: преобразовании неявных знаний в явные; поиске пробелов в знаниях; извлечения и интеграции знаний из различных источников; изучении распределенных Web-систем; анализе неструктурированной информации – текстов, изображений, экранов и др.

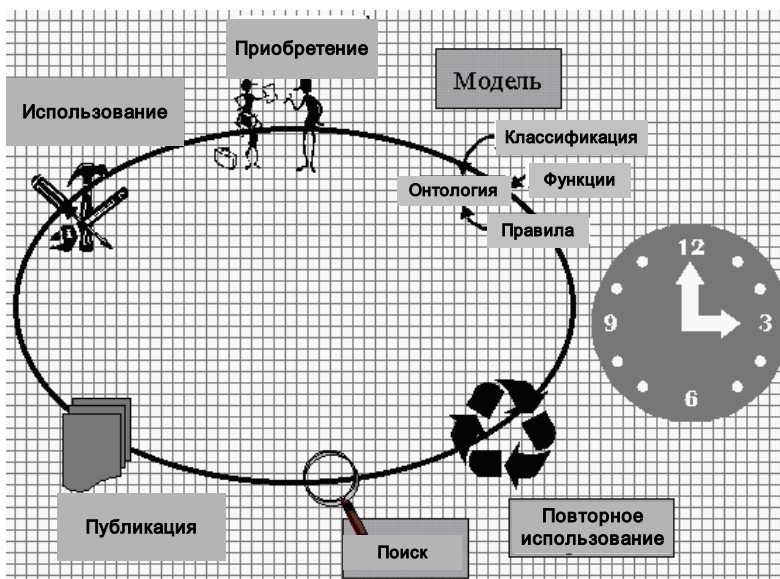


Рис. 2. Модель жизненного цикла знаний

2. *Моделирование знаний* (knowledge modelling) состоит в рассмотрении извлеченных знаний и их использовании. При моделировании знаний в виде модели, она должна быть пригодной для хранения, накопления и применения знаний при решении конкретных задач.

3. *Поиск и выборка знаний* (knowledge retrieval). Накопленные знания используются при эффективном поиске ранее сохраненных знаний и подмножеств контентов, релевантных решаемой задаче.

4. *Повторное использование знаний* (knowledge reuse). Выявляемые знания сохраняются в определенном формате в базах знаний для многократного применения в разнообразных контекстах разных предметных областей.

5. *Публикация (распространение) знаний* (knowledge publishing). Цель публикации – предоставление знаний в удобной стандартной форме, понятной пользователям.

6. *Использование и обновление знаний* (knowledge maintenance). Использование знаний может повлечь за собой глубокий анализ содержания (контента) знаний и требовать его обновления. Контент знаний должен подвергаться верификации и валидации, а также сертификации для сохранения и использования другими.

1.1. Управление знаниями

Для определения целей и задач любой предметной области используется подход, именуемый управлением научными знаниями. Объектами исследования предметной области знаний являются готовые ресурсы сети, описанные в машинной семантике. С помощью машинно-читаемых языков (machine-readable languages) задается взаимодействие и интеграция гетерогенной информации в глобальной информационной сети. Семантика ресурсов глобальной сети, включая сервисы, задается на уровнях исследования [7-11]:

1. Верхний уровень обеспечивает уникальную глобальную *идентификацию* ресурсов, описание *метаданных* о ресурсах средствами *языков описания метаданных* и знаний, которые являются базовыми понятиями *онтологий*, обеспечивают взаимопонимание и задание общепринятого словаря для метаданных и механизмов вывода новых знаний.

2. Базовый уровень сервиса знаний включают стандартные умозаключения, целенаправленный просмотр метаданных онтологий, а также дает объяснение выполненным умозаключениям. К этому уровню относятся сервисы:

- поддержки методов извлечения знаний по прецедентам (Case-based reasoning, CBR);
- кластеризации, индексации, уточнения и визуализации больших объемов информации;
- связи одних наборов онтологий с другими и с заданной концептуальной схемой онтологии;
- обеспечения задач умозаключения, касающихся мониторинга, диагностики причин появления исключительных ситуаций и оценивания их завершения.

При обработке знаний сервис формирует *управляемые словари* (controlled vocabularies), в которых отражены области знаний о предметной области и возможности машинных сервисов. Обработка знаний основана на *логическом выводе* (inference engines) или на *дескриптивной логике* (Description Logics, DL). Представление знаний о прикладной предметной области и формулировка высказываний осуществляется с помощью логики предикатов первого порядка и средств языков описания онтологии OWL – OWL-DL и OWL-Lite [5, 6, 12].

1.2. Онтология представления знаний

Понятие онтологии, заимствованное из философии, сейчас используется в искусственном интеллекте и информатике. В искусственном интеллекте онтологии определяются в контексте концептуализации знаний и систем, основанных на знаниях. В некоторых работах онтологии описываются на основе понятий и конструкций логики и математики. К настоящему времени построено много различных онтологий и область их применения увеличивается.

Основу онтологии составляют понятия и термины, которые включаются в словарь терминов [10-14]. На формальном уровне онтология это система, состоящая из набора понятий и набора утверждений об этих понятиях, на основе которых можно строить классы, объекты, отношения, функции и теории. Онтология в Web задает *семантику* на уровнях знаний, с помощью которой уточняются, обогащаются и аннотируются контентные глобальной сети. Использование средств онтологии способствует повышению качества коммуникаций между системами, машинами, пользователями и организациями. Эти средства используются для:

- аннотирования прикладных (предметных, проблемных) областей (ПрО), например, моделей ПрО, оформления терминологии в ПрО и др.;
- описания взаимозависимости задач и процессов научных исследований в ПрО;
- аннотирования характеристик качества результатов исследований (качества знаний);
- персонификации (создание моделей для потребителей знаний);
- аргументации поиска знаний и моделирования контента.

Контент предметной области в онтологии, может использоваться в *технологиях знаний*, которые обеспечивают фильтрацию информации и повышение уровня интеллектуальности технологий. Технология знаний поддерживается множеством инструментов, включающих языки и средства представления семантики ПО, язык управления контентом и обмена информацией и знаниями (рис. 3) [1-6].

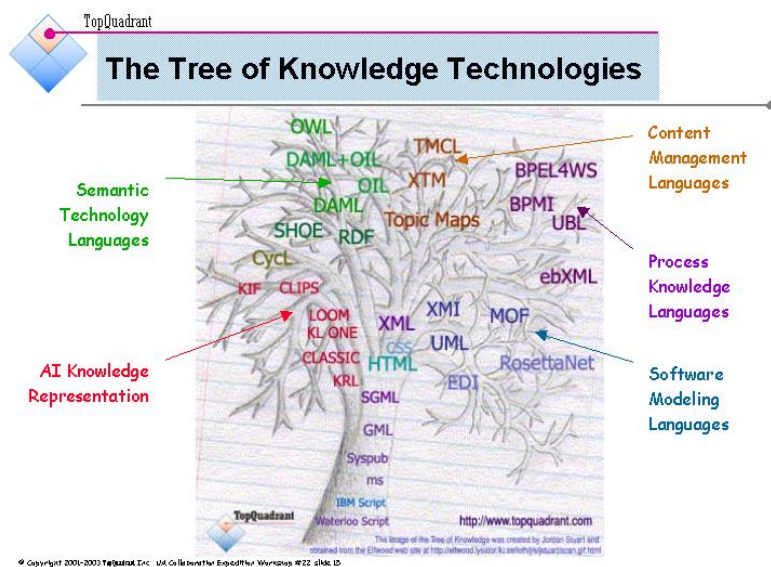


Рис. 3. Дерево технологии знаний²

² Рисунок взят на сайте <http://www.coolheads.com/egov/combined/topicmap/s167/img15.html>

Для построения онтологий используется инструмент *Protégé* (<http://protege.stanford.edu/>), который поддерживает два основных способа моделирования онтологий – с помощью редакторов Protégé-Frames и Protégé-OWL. Онтологии могут экспортироваться в разные форматы, включая RDF(S), OWL и XML Schema. В основу подмножества языка описания онтологий OWL (Ontology Web Language) и языков OWL-DL и OWL-Lite положена дескриптивная логика [3, 4].

К современным инструментам представления знаний и рассуждений в терминах дескриптивной логики относятся: *FaCT* (на LISP), *FaCT++* (на C++), *CEL* (2005), *KAON2* и *Pellet* (на Java), *MSPASS* (на C) и др. Необходимой составляющей инфраструктуры уровня знаний являются *прикладные сервисы*, ориентированные на коллективное решение задач (problem-solving environments, PSE) научным сообществом.

1.3. Средства создания моделей знаний

Процесс создания и документирования *моделей знаний* задается методологиями, основанными на знаниях (Knowledge-based systems, KBS), в том числе методологией CommonKADS [10], которая базируется на идее построения библиотек, содержащих элементы решения задач в проблемных областях, являющиеся повторно использованными (reuses) в других областях. Данные инструменты моделирования расширены агентной методологией для проведения мультиагентного анализа и проектирования систем, основанных на знаниях. В этих инструментах определяется множество моделей, охватывающих аспекты процесса разработки *систем управления знаниями*.

К моделям знаний, которые создаются, относятся следующие:

- *модель агентов* для спецификации характеристик, определяющих способность агента к рассуждению, сенсорным /эффекторам (sensor/effectors) и сервисам;
- *модель задач*, которые могут выполняться агентами для достижения поставленных целей;
- *организационная модель*, которая задает социальную структуру сообщества агентов;
- *координационная модель*, задающая переговоры (общение) между агентами;
- *коммуникационная модель*, детализирующая взаимодействие агентов, людей и программных агентов;
- *модель проекта* системы включает типичные действия по проектированию, связанному с определением сети агентов, выбора наиболее подходящей архитектуры агентов и платформы для разработки новых агентов.

Единую *платформу знаний* на домене (Knowledge Domain) формирует инструмент *GRACE* (<http://www.grace-ist.org/>). С его помощью задаются такие аспекты, которые определяют ресурсы информации, онтологии и сообщества. Домен знаний представляется в виде концептуального фрейма, содержащего документы описаний, образующих единую структуру представления домена. GRACE обеспечивает предварительную индексацию документов из многих источников контента с помощью *инженерии онтологий*, описываемой ниже [5, 6].

1.3.1. Инженерия онтологий

Онтология, как инструмент представления знаний о предметной области, задает инженерные аспекты обработки онтологического описания [9], выполняя:

1. задание основной онтологии в виде запросов, которые систематически передаются множеству источников контента для выбора подходящих документов (эти запросы повторяются периодически, что позволяет обеспечить обновление документов, созданных средствами GRACE);
2. отбор документов для анализа содержания и связи с различными понятиями онтологии;
3. просмотр и выполнение запросов с индексами, которые отмечают документы источника контента, из которого они были выбраны.

Для выборки информации из онтологии используется система Grid (в частности, Data Grid). Публикация и распространение знаний поддерживается рядом *систем управления документами* (document management systems), которые используют языки разметки контента для индексирования, поиска и предоставления контента. Некоторые из них используют персонализацию (customization or personalization) контента в системе разработки АКТ (Advanced Knowledge Technologies, www.aktors.org). Они позволяют персонафицировать контент и представлять его в соответствии с интересами пользователей онтологии (<http://eldora.open.ac.uk/my-planet/>).

Для публикации знаний используется *ePrints* (<http://www.eprints.org>), который предоставляет набор сервисов для архивирования публикации и сопровождения их расширенными метаданными. Это способствует созданию разнообразных сервисных знаний, основанных на методах контентного поиска (content-based retrieval).

В развитие инструментов, поддерживающих создание электронных библиотек, и Grid-технологий среды e-science, в проекте *TextGrid* разработаны **текстовые решетки Grid** [7, 11]. С их помощью предоставляются средств обработки, анализа, индексирования, аннотирования, редактирования и опубликования текстовых данных для академических исследований.

Для решения проблемы упорядочения текстовой информации используется *система управления терминологией* (System Quirk – SQ, <http://www.computing.surrey.ac.uk/SystemQ>). Эта система поддерживает создание и ведение терминов в терминологической базе данных, а также организацию коллекций текстов на компьютере. Система включает в себя целый набор разнообразных инструментов: (*Virtual Corpus, KonText, Ferret* и др.).

В функции инфраструктуры уровня знаний входит также *поддержка коммуникаций* при совместной (коллективной) работе сообщества ученых. Эти функции реализуются набором средств организации *Web-конференций* и *виртуальных распределенных рабочих пространств* (virtual shared workspace), которые в значительной мере облегчают диалог и коммуникации между отдельными лицами и группами, как, например, среда CAVE [9-11].

Один из базовых компонентов, которыми должны оснащаться региональные центры UK (e-Science Regional Centres), – **сети доступа** (access grids). Эти сети поддерживают распределенные заседания, сессии коллективного творчества, семинары, лектории и тренинги. Сеть доступа – это коллекция ресурсов и технологий, которые обеспечивают аудио- и видео-сотрудничество в территориально распределенных сообществах. Архитектура сети доступа включает широкоформатные мультимедийные дисплеи, презентации и интерактивные среды, а также интерфейсы с программным обеспечением промежуточного слоя сети Grid и сред визуализации.

Инструментальную поддержку функционирования сети доступа осуществляет **Access Grid®**, текущая версия которого – *Access Grid 3.1 beta1* – базируется на стандартизованных Интернет-технологиях и протоколах. При этом все сетевые соединения защищены и сертифицированы [13, 14].

1.3.2. Жизненный цикл онтологии

Жизненный цикл онтологии включает следующие действия [11]:

- Поддержка жизненного цикла знаний. Базовые концепции и языки проекта Semantic Web, в целом пригодные для того, чтобы специфицировать и поставлять сервисы на информационном уровне. Они являются основой для моделирования *знаний* и выполнения рассуждений с использованием знаний.
- Достижение доверия к добываемым знаниям. Проблема доверия к результатам логического вывода связана с обеспечением качества информации и логического вывода.
- Обеспечение качества информации, используемой машинами вывода. Информацию необходимо проверять на *корректность* и *полноту* фактов, содержащихся в аннотациях.
- Обеспечение качества машин логического вывода. Машины логического вывода (для сервисов базового уровня) зависят от количества и качества аннотаций, найденных в обогащенном контенте. Требуется определить, насколько надежными являются методы логического вывода.

Модель жизненного цикла знаний включает основные, вспомогательные и организационные процессы (аналогично стандарта ISO/IEC Life Cycle 12207). К вспомогательным процессам можно отнести извлечение знаний для построения онтологии, оценивание онтологий, верификацию, интеграцию и документирование онтологии. К организационным процессам относятся процессы планирования построения онтологии и обеспечения гарантии качества онтологий (согласованность, полнота, лаконичность и др.).

1.3.3. Виды онтологий

К видам онтологий для проблемных областей относятся:

- *онтология ПрО* – объекты, свойства и их взаимосвязи в домене. По данным в [1] разработано множества аннотаций для медицинских образов, аннотаций для информации о климате, словарях для описания важных характеристик в инженерном проектировании и др.;
- *онтология задач в ПрО* - это концептуализация задач и процессов, их взаимозависимости и свойства;
- *онтология программного обеспечения* (готовых ресурсов) ПрО для повторного использования отдельных элементов;
- *онтология сообщества специалистов* в ПрО и др.

К настоящему времени разработаны стандартные методы и средства *верификации* онтологий, *измерения* (метрик), моделирования и оценивания качества онтологий, а также затрат на построение онтологий.

1.3.4. Задачи онтологий

К основным задачам онтологии относятся следующие [12]:

- Широкомасштабное аннотирование и обогащение онтологии *существующими* научными данными, информацией и знаниями для различных предметных областей.

- *Разработка сервисов онтологий.* Сервисы предоставляют согласованные словари и концептуальные научные области (домены) для поддержки аннотирования и извлечения знаний. Концептуальные описания включают понятия и отношения, возникающие между объектами и процессами, представляющими интерес для исследователей в соответствующих областях знаний. Сервисы онтологий управляют отображением онтологии, которое используется агентами с различными интересами, а также сервисы онтологий для построения онтологий по XML-схемам.
- *Разработка сервисов персонификации.* Эти сервисы обеспечивают взаимодействие с сервисами аннотирования и сервисами онтологий.
- *Разработка агентов.* При разработке агентов проводится учет (accounting), мобильности агентов и обеспечение безопасности (security) [13]. Предотвращение неавторизованного манипулирования на удаленных Grid-узлах, и наоборот, защита Grid-узлов от неавторизованного манипулирования чужими агентами (не ограничивая гибкости Grid).

2. Web-средства и Web-Grid в электронной науке

Web-средства накладывают определенные ограничения на инфраструктуру и возможности систем электронной науки. Типичная распределенная транзакция предусмотрена для большинства сетевых приложений. Поток информации всегда инициируется клиентом. Если приложение получает информацию спонтанно от сервера HTTP, необходимо включить функциональные возможности сервера HTTP. Типичные сетевые транзакции охватывают намного меньшее число машин, чем того требуют Grid-вычисления. Из-за технических ограничений инфраструктуры глобальной информационной сети семантический Web не может обеспечить пользователей электронной науки (e-ученых) правильной информацией в нужный момент времени. Эта проблема усугубляется все возрастающими объемами информации, которые будут генерироваться в этих системах. В связи с этим возрастает потребность в вовлечении в e-science *Grid-технологий* и развитии семантического Web в направлении *Семантического Grid*. Это даст возможность облегчить гибкое сотрудничество и проводить вычисления в глобальном масштабе с помощью системы Grid [16-20].

2.1. Сервис для описания Web-систем

Сервисы Интернет обеспечивают решение разного рода прикладных и бизнес-задач [14, 20, 21]. К сервисам относятся:

- общие сервисы системных сред для поддержки процессов и функций обработки программ и данных (например, службы именования, каталогизации и др.);
- сетевые сервисы стандартной модели OSI, моделей SOA (Service-oriented Architecture), SCA (Service-Component Architecture), как инструменты представления и обработки сервисов в сети Интернет, которые реализуют деловые, финансовые, экономические и другие услуги при решении соответствующих задач;
- готовые программные и информационные ресурсы (services, artifacts, reuses, assets и др.), используемые как многократно, повторно используемые сервисы для решения разного рода задач в e-science и других прикладных областях.

Некоторые из сервисов стали обязательной частью общесистемных средств (VS.Net, IBM, Intel, Linux и др.), другие используются в специальных областях (например, медицина, биология) в плане предоставления сервисов при работе с современными данными (FDT, GDT, Big Data).

2.2. Инструменты поддержки сервисов

Для проведения композиции сервисов предлагается инструмент – *Jopera for Eclipse* (<http://www.jopera.ethz.ch/>) [21-22]. Этот инструмент предоставляет пользователю визуальный язык и гибкую исполнительную платформу для построения распределенных приложений на базе повторно используемых сервисов, в том числе Web-сервисов. В сети Интернет имеются мощные интегрированные инструменты для:

- визуального определения процесса (потока управления и потока данных);
- облегченной (agile) композиции сервисов с немедленной обратной связью, рефакторингом и регрессионным тестированием процессов;
- рекурсивной композиции сервисов путем автоматического опубликования процессов как Web-сервисов;
- эффективного исполнения процесса с помощью визуальных моделей, которые компилируются в исполнительный код Java;
- визуального мониторинга, интерактивного управления и бесшовной (seamless) отладки композиций сервисов, интегрированных в инструменты проектирования;
- управления изменением интерфейсов сервиса при условии, что JOpera сообщает, каких процессов будут касаться изменения в сервисе;
- трассирования «родословной», т.е. метаданных, содержащих сведения о происхождении данных, которые автоматически собираются движком WF;
- масштабируемого и автономного исполнения процесса с помощью версии JOpera, обеспечивающей запуск кластеризованных компьютеров.

JOpera обеспечивает набор Eclipse-плагинов для связи различных программных компонентов, допускает итеративную композицию сервисов, гетерогенность сервисов (включая маршрутизаторы SOAP и RESTful Web-сервисы, Grid-сервисы, Java snippets и др.), а также моделирование и исполнение процесса. Версия *JOpera 2.1.2* находится по адресу <http://www.jopera.ethz.ch/download/index>.

2.3. Поиск сервисов

Для онтологического поиска сервисов по их семантическим описаниям используется компонент *Feta*, состоящий из *Feta Client* и *Feta Engine*. *Feta Client* – это GUI-плагин к инструменту Taverna, используемый для описания сервиса, а *Feta Engine* – Web-сервис.

Подключаясь к *Feta Engine*, плагин *Taverna Feta* позволяет:

- конструировать ориентированные на ПрО семантические запросы к нужным сервисам, которые должны отсылаться на *Feta Engine*;
- отображать информацию о результатах выполнения запроса на поиск сервисов;
- интегрировать результирующие сервисы в WF.

Критериями поиска сервисов являются:

- сервис, на входе которого приемлем элемент семантического или общего типа X;
- сервис, который производит на выходе элемент семантического типа Y;
- сервис, который решает задачу X или еще более конкретную;
- сервис, который использует метод X или еще более конкретный;

- сервис определенного типа, например, процессор WSDL и др.

Фрагмент формы для составления запросов на поиск сервисов в среде Taverna Workbench представлен на рис. 4. Пример отображения результатов поиска представлен на рис. 5.

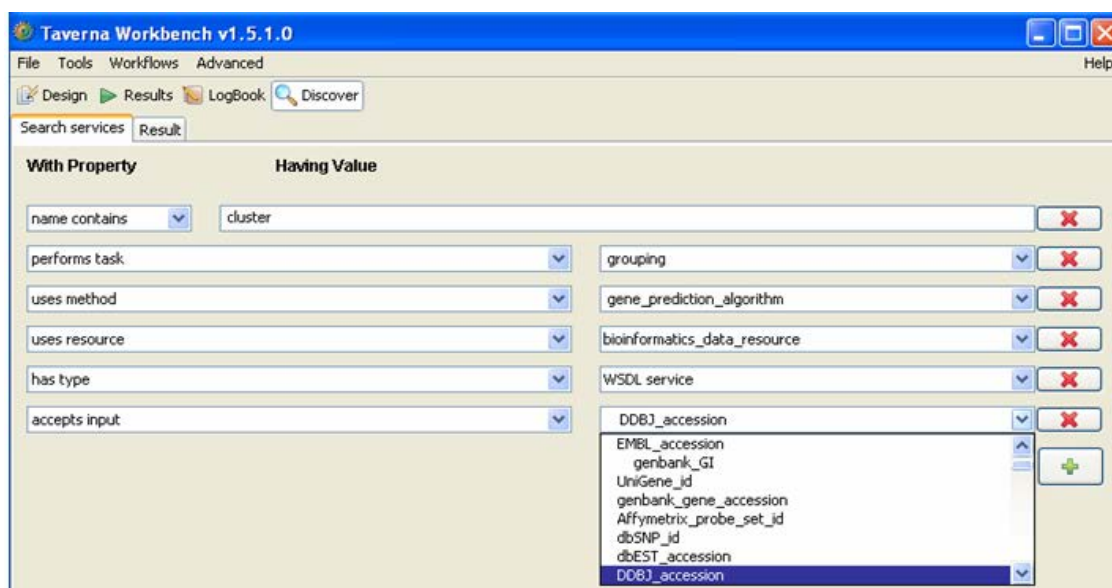


Рис. 4. Форма для задания критериев онтологического поиска сервисов.

3. Система Grid и Web-сервисы

Система Grid обеспечивает распределенные вычисления, в которой «виртуальный суперкомпьютер» представлен в виде кластеров, соединённых с помощью сети слабосвязанных гетерогенных компьютеров, работающих вместе для выполнения огромного количества заданий (операций). Для сетевых и распределенных вычислений используется система BOINC (Berkeley Open Infrastructure for Network Computing), разработанная в университете Беркли. Данный инструмент используется [13]:

- в астрофизике, гравитационной физике, физике высоких энергий, физике нейтрино и ядерной физике;
- молекулярной динамике, информатике и вычислительной технике, нанотехнологии;
- структурной биологии, вычислительной биологии, геномике, протеомике и медицине.

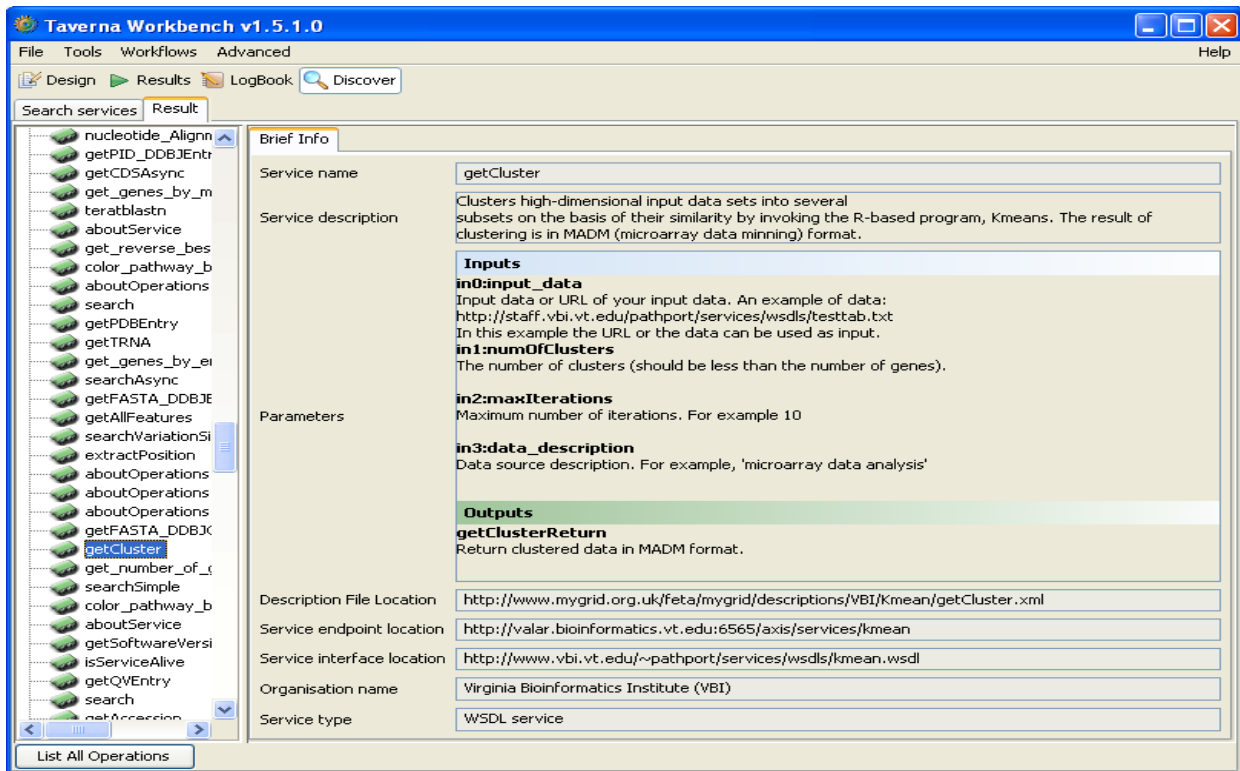


Рис. 5. Результат онтологического поиска сервиса по критерию.

Система OGSA (Basic Execution Service) определяет интерфейс к сервисам, которые инициируют вычислительные процессы в Grid, отслеживает и управляет вычислительной активностью и моделями жизненного цикла (состоянию) процессов, а также информационными модели вычислительного процесса. Ws-naming определяет схему для ссылки на ресурсы в Grid на абстрактном уровне, независимо от их физического расположения. Ws-dissemination определяет сервисы для распространения сообщений в Grid (рис.6) среде.

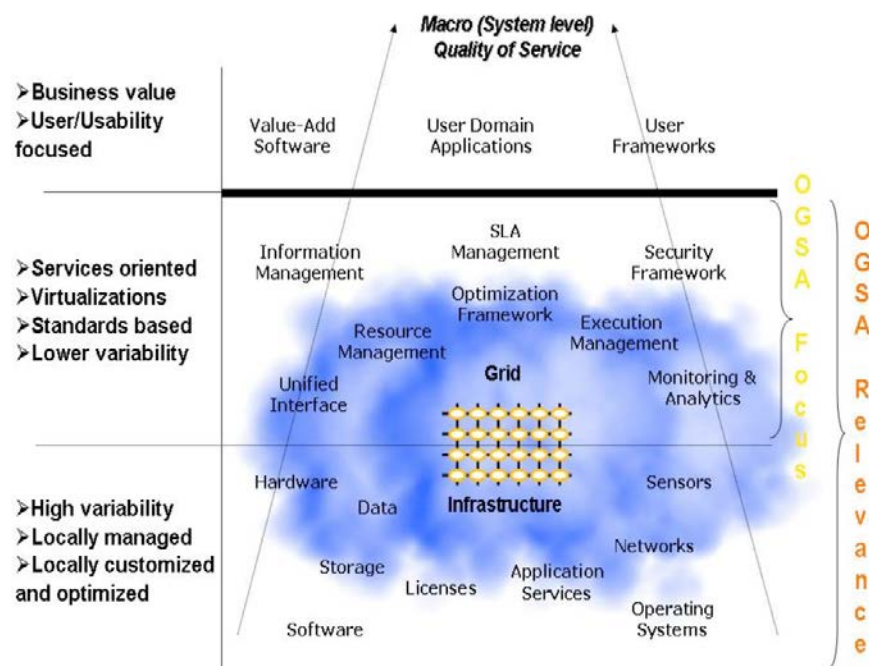


Рис.6. Схема Grid-системы

Web-сервисы *не имеют состояния*, и срок их жизни определяется поставщиком услуги, то есть они существуют независимо от потребителя [13-16].

Grid-сервисы имеют состояние и срок жизни, ограниченный *потребностями потребителя*. Они «рождаются» на серверах в Grid-сети для решения конкретной задачи и «исчезают» по ее завершении. Представляют «свой» сервер не как что-то, реально оказывающее услуги, а как *фабрику*, на которой по заказу клиента создается нужный сервис, используемый в заданное время. Этот подход получил название «фабрики образцов».

Если возникает необходимость решить какую-то математическую задачу в Grid-сети, под нее создается виртуальный компьютер, предоставляющий нужный *физический сервер*. Этот сервер может поддерживать множество подобных *виртуальных вычислительных машин*, создаваемых под требования клиента и уничтожаемых, когда в них отпадает необходимость.

Для представления спецификаций Grid-сервисов и других ресурсов используются **WSRF** (*Web Services Resource Framework*) (<http://toolkit.globus.org/wsrf/>). С помощью средств спецификации проводится моделирование и управление состоянием контекста Web-сервисов. Это дает возможность контролировать и изменять состояние доступных сервисных ресурсов (*WS-Resource*).

3.1. Grid-архитектура

Grid-архитектура представляет собой архитектуру взаимодействующих *протоколов, сервисов и интерфейсов*. Эти элементы сохраняются в репозитории. Ими управляет брокер сервисов. С помощью сервисов и ресурсов устанавливается соединение с Grid-системой для вычисления различного рода задач. Для обработки ресурсов выполняются протоколы следующих уровней (рис. 7) [11]:

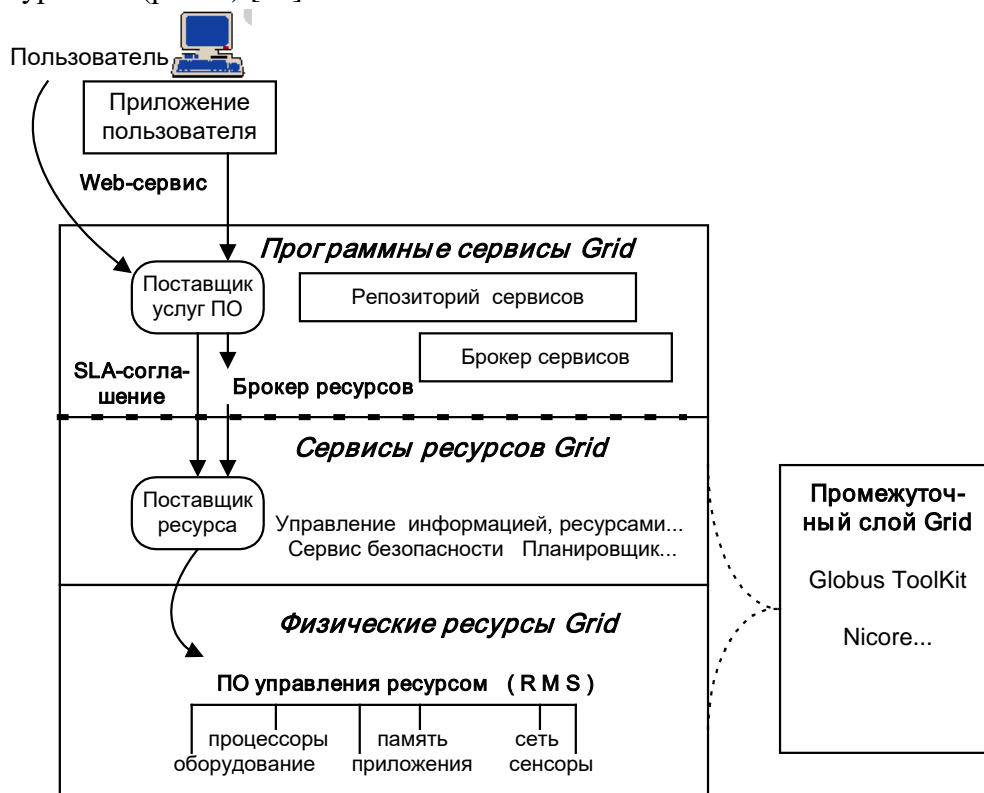


Рис. 7. Архитектура Grid

Уровень ресурсов Grid содержит физические ресурсы, к которым нужно предоставлять совместный доступ (ресурсы вычислений (процессоры), ресурсы памяти, сетевые ресурсы). Физические ресурсы обладают всеми характеристиками оборудования Grid – многочисленность, гетерогенность конфигурации и разнообразие политик использования. На каждом месте размещения ресурсов ими управляет *программное обеспечение управления ресурсом* (resource management software (RMS)), которое решает задачи управления – планирование, распределение, контроль *состояния ресурсов* и предоставление доступа пользователям.

Уровень услуг (сервисов) ресурсов Grid предоставляет сервис планирования ресурса и вычислений. В первом случае ресурсы предоставляются на определенный период времени. Во втором случае сервисы выполняют интенсивные вычисления и возвращают результаты. Для получения ресурсов Grid работает брокер ресурсов.

В основе Grid лежит взаимодействие протоколов, сервисов и интерфейсов, определяющие базовые механизмы и устанавливают соединения с Grid-системой для совместного использования ресурсов.

Архитектура протоколов Grid включает уровни (рис. 8), смысл которых дан ниже.



Рис. 8. Уровни архитектуры протоколов Grid и Интернет

Базовый уровень (Fabric Layer) описывает сервисы, непосредственно работающие с ресурсами, ресурсами памяти и информационными ресурсами.

Сетевой ресурс – связующее звено между распределенными ресурсами Grid. Его основная характеристика - скорость передачи данных.

Связывающий уровень (Connectivity Layer) определяет коммуникационные протоколы и протоколы аутентификации. *Коммуникационные протоколы* обеспечивают обмен данными между компонентами базового уровня (ресурсами). *Протоколы аутентификации* основываются на коммуникационных протоколах и предоставляют криптографические механизмы для идентификации и проверки подлинности пользователей и ресурсов.

Ресурсный уровень (Resource Layer) работает над протоколами коммуникации и аутентификации уровня в архитектуре Grid. Этот уровень реализует инициацию, мониторинг и контроль ресурсов. Он обеспечивает взаимодействие с ресурсами с помощью *унифицированного интерфейса*.

Коллективный уровень (Collective Layer) отвечает за *глобальную интеграцию различных наборов ресурсов*.

3.2. Инструменты для разработки Grid-систем

Для разработки Grid-систем используются следующие инструменты [18-22]:

- **Legion** (<http://legion.virginia.edu/>) обеспечивает объектно-ориентированную модель инфраструктуры, где все ресурсы описаны в виде объектов, и предоставляет систему единой целостной виртуальной машины;
- **Globus Toolkit** (GGF) обеспечивает сервисно-ориентированную слоистую архитектуру, в которой глобальные сервисы строятся на основе базовых сервисов нижележащих уровней;
- **Condor** (<http://www.cs.wisc.edu/condor>) – открытый проект, разработанный для выполнения заданий с интенсивными вычислениями, не требующими вмешательства человека. Задание специфицируется в файле конфигурации и его выполнение распределяется по множеству узлов Grid-сети. Данный инструмент использует *Globus Toolkit* для управления заданиями;
- **WebFlow** (<http://www.npac.syr.edu/users/haupt/WebFlow/>) обеспечивает создание целостной оболочки с возможностью публикации и многократного использования вычислительных модулей, чтобы пользователи могли работать с Web-браузером в конструировании распределенных приложений и с редактором в качестве визуального инструментального средств. Нижний (серверный) слой системы реализован с помощью *Globus Toolkit*;
- **Gridbus Data Grid Service Broker** развивает модель брокера ресурсов вычислительного Grid, в плане использования распределенных сетей, ориентированных на данные; динамическую параметризацию для доступа к репозиториям и оптимизации передачи данных;
- **GRACE** (Grid Architecture for Computational Economy) – развитие *Nimrod/G*. Инструмент обеспечивает динамическое сотрудничество с владельцами ресурса для выбора тех ресурсов, которые предлагают *оптимальную стоимость* использования или критериям оптимального времени;
- **Grid-порталы** – позволяют обращаться к ресурсам, определенным для конкретной ПрО, через Web-интерфейс, и обеспечивают доступ к Grid-ресурсам. Доступ к Grid-порталу может также индивидуализироваться при помощи профилей, которые создаются и сохраняются для каждого пользователя (например, портал *HotPage*);
- **Grid Port Toolkit** – порталный набор инструментов многократного использования, основанный на инфраструктуре *HotPage*. В нем два компонента: *GridPort* – сервисы Grid-портала и прикладной программный интерфейс (API) приложений. Модуль Web-портала выполняется на сервере сети и обеспечивает безопасное подключение к Grid. API приложений обеспечивает Web-интерфейс, который помогает конечным пользователям разрабатывать специализированные порталы (не требуя знаний об основной порталной инфраструктуре). *GridPort* спроектирован так, чтобы позволить выполнение порталных сервисов и приложений клиентов на отдельных Web-серверах. Портальная архитектура основана на трехслойной модели, где браузер клиента надежно подключается к Web-серверу по безопасному соединению через двунаправленный канал (по протоколу HTTP). Web-сервер может получить доступ к различным Grid-сервисам, используя инфраструктуру *Globus*.

4. Направления развития е-науки на перспективу

К направлениям развития е-науки относятся [7-21, 22]:

1. Информатизация е-науки, включающая:
 - электронные библиотеки, документооборот, регламентация доступа к данным, защита информации (данных, знаний) и др.;

- широкомасштабное аннотирование и обогащение онтологиями существующих научных данных, информации и знаний в разных областях, в частности построение онтологий для специалистов областей и разработанных ПС для отдельных предметных областей.
2. Семантический Web:
- совершенствование методов и инструментов аннотирования, интеграции данных и информации, извлечения (приобретения) новых знаний, сервисов онтологий и т.п.;
 - стандартизация процесса построения, интеграции, уточнения, верификации и измерения онтологий и др.
 - обеспечение качества информации путем верификации, тестирования и оценки Web-сервисов.
3. Развитие семантического Grid в направлении:
- виртуализации ресурсов;
 - разработки методов проектирования и развертывания приложений в Grid-средах;
 - создания средств взаимодействия ученых мира и определения форм сотрудничества в Grid-среде;
 - оснащение Grid-среды средствами проведения распределенных вычислений и др.
4. Потоки научной работы (Scientific workflow).
- определение организационных процессов в е-науке;
 - создание систем управления научными процессами;
 - разработка методов моделирования потоков работ (прототипирование, метамоделирование и др.);
 - разработка методов верификации и валидации потоков работы;
 - разработка методов анализа эффективности и качества систем управления потоками работ.

К настоящему моменту приведенные пути развития е-науки получили широкое развитие.

5. Заключение

В работе дано определение целей и задач электронной науки, предназначенной для проведения научных экспериментов с использованием механизмов представления знаний и онтологий, а также механизмов Web-Grid. Основу этой науки составляют знания, информация и онтология. Приведена принятая в е-науке схема процесса представления знаний для любой предметной области. Рассмотрены структура моделей знаний, онтологии, инженерии концептуализации знаний и задач исследования научных знаний в разных научных областях. Рассмотрены Web-средства и инструменты для создания из готовых ресурсов и сервисов Grid-систем. Приведена архитектура Web-Grid (репозиторий, ресурсы, сервисы, брокер ресурсов и др.), семантического Grid и Web, стандарты протоколов обработки сообщений при работе и выполнении инструментов для создания Web-систем. Определены перспективы развития е-науки, в том числе и потоки научных работ (Scientific workflow).

6. Литература

1. Bohle, S. What is E-science and How Should it Be Managed? http://www.scilogs.com/scientific_and_medical_libraries/what-is-e-science-and-how-should-it-be-managed/

2. IEEE International Conference on e-Science, homepage, accessed December 18, 2014, <https://escience-conference.org/>
3. Science 2.0: новая научная парадигма или развитие инструментария исследовательской работы .- А.С.Милославов //Сборник научных статей XVIII Объединенной конференции «Интернет и современное общество» IMS-2015, Санкт-Петербург, 23-25 июня 2015.
4. European organization for nuclear research CERN.
<http://public.web.cern.ch/public/en/research/DataAnalysis-en.html>
5. OWL Web Ontology Language. W3C Working Draft 29 July 2002. Latest version is available at <http://www.w3.org/>
6. OWL-язык Веб-онтологий. http://sherdim.ru/pts/semantic_web/REC-owl-features-20040210_ru.html
7. De Roure D., Jennings N. R., Shadbolt N. R. The Semantic Grid: Past, Present and Future e-Science Infrastructure // <http://www.semanticgrid.org/documents/semgrid-journal/semgrid-journal.pdf>.
8. Semantic Web, Representation of data on the World Wide Web, based on the RDF standards, <http://www.w3.org/2001/sw/>
9. TopQuadrant Technology Briefing. Semantic Technology. Version 1.2 (march 2004) // http://www.topquadrant.com/documents/TQ04_Semantic_Technology_Briefing.PDF
10. Semantic Web programming//S. Hebler, M. Fisher, R.Blace, A/Peter-Lopes, Willey Publishing Inc., 2008 <http://semwebprogramming.org>
11. Goble C., Bechhofer S. OntoGrid: A Semantic Grid Reference Architecture // <http://www.ctwatch.org/quarterly/articles/2005/11/ontogrid-a-semantic-grid-reference-architecture/>
12. W3C OWL Working Group OWL2 Web Ontology Language, W3C.2009 <http://www.w3.org/TR/owl2-overview/>
13. Демичев А. П., Ильин В. А., Крюков А. П., Введение в грид-технологии. Препринт НИИЯФ МГУ, 2007, 11/832 2007, <http://www.sinp.msu.ru>.
14. Лаврищева Е. М., Карпов Л. Е., Томилин А. Н., Семантические ресурсы для разработки онтологии научной и инженерной предметных областей, Доклад на конференции "Научный сервис в сети Интернет" 19-24 сентября 2016, Абрау.
15. SOAP Version 1.2 Part 1: Messaging Framework (Second Edition) W3C Recommendation 27 April 2007 <http://www.w3.org/TR/SOAP12-part1/>
16. Web Services Description Language (WSDL) 1.1 W3C Note 15 March 2001 <http://www.w3.org/TR/wsdl>
17. W3C. RDF Current Status. <http://www.w3.org/standards/techs/rdf>
18. SPARQL Query Language for RDF W3C. <http://www.w3.org/TR/rdf-sparql-query/>
19. W3C Annotea Project. <http://www.w3.org/2001/Annotea/>
20. Reference Model for Service Oriented Architecture 1.0 <http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.html>
21. Web Services Resource 1.2 (WS-Resource) http://docs.oasis-open.org/wsrf/wsrf-ws_resource-1.2-spec-os.pdf
22. Web Services Resource Properties 1.2 (WS-ResourceProperties) http://docs.oasis-open.org/wsrf/wsrf-ws_resource-1.2-spec-os.pdf
<http://docs.oasis-open.org/dita/v1.2/spec/DITA1.2-spec.html>