

P10-2003-140

В. В. Галактионов

**WEB SERVICES — СЕРВИС-ОРИЕНТИРОВАННАЯ
ТЕХНОЛОГИЯ ДЛЯ РАСПРЕДЕЛЕННЫХ
ОБЪЕКТНЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ.
ОСНОВНЫЕ КОНЦЕПЦИИ, ПРОТОКОЛЫ
И СПЕЦИФИКАЦИИ**

Введение

Web Services – новая технология для *развертывания* распределенных вычислительных систем. Основная причина ее появления – неспособность существующих технологий, таких как объектные системы типа COM семейства Microsoft и стандарты OMG CORBA, в полной мере обеспечить совместимость (интероперабельность) различных программных продуктов для неоднородных распределенных систем. Требование совместимости – один из главных рычагов давления все стремительно развивающегося рынка электронного бизнеса, основу которого составляют многочисленные разнотипные как поставщики вычислительных услуг, так и их потребители. Технология Web Services предложена и развивается под эгидой WWW-консорциума W3C [1] ведущих компаний-производителей программного обеспечения.

Основу ее составляют:

- простейшие коммуникационные Интернет-протоколы **HTTP** и/или **SMTP**;
- протоколы **SOAP** (Simple Object Access Protocol) для управления сообщениями в универсальном XML-формате;
- язык **WSDL** (Web Services Definition Language) описания интерфейса взаимодействия компонент распределенной системы.

Технологии услуг сети SOAP и WSDL применялись и ранее (Userland, Microsoft, Developermentor) вне W3C; их спецификацией были использованы для Web Services как основание для создания расширяемой структуры управления сообщениями (messaging) SOAP 1.2 и языка определений интерфейсов WSDL 1.2.

Можно дать определение Web Services (или Web-служб) как набора услуг в виде программных приложений, идентифицированного сетевым адресом URI (Uniform Resource Identifier), интерфейсы и *связывания* (binding) которого определяются XML-средствами. Web-службы обеспечивают прямые взаимодействия через Интернет с другими агентами программного обеспечения, используя сообщения, основанные также на XML-формате. Данное определение Web-служб не предполагает использование SOAP в качестве формата или модели обработки сообщений. И при этом оно не предполагает также использование WSDL как языка описаний обслуживания. В настоящее время существует (и будет появляться) множество услуг сети, которые используют уровень HTTP в качестве протокола передачи данных и некоторые взаимосогласованные XML-форматы содержания сообщения (контента). Однако в W3C предполагают, что более высокие уровни стека протокола Web-служб должны строиться на основе SOAP и WSDL.

Консорциум W3C вырабатывает рекомендации, являющиеся стандартом для спецификаций и протоколов Web-служб; согласно этим рекомендациям различные компании-производители программного обеспечения разрабатывают собственные реализации этих технологий. Вот некоторые из них: SOAP Toolkit (Microsoft), WebSphere Application Server (IBM), JWSDP (Sun). Одним из замечательных достижений технологии Web-служб является совместимость всех их реализаций, не зависящая от поставщиков (провайдеров) вычислительных услуг и производящих их технологий. Основу этой совместимости составляет последовательное применение на всех уровнях предоставления услуг Web-служб стандартов XML-технологии. Так например, формат SOAP-сообщений – основной

единицы передачи данных - представлен в виде XML-документа; описание интерфейса вычислительного сервиса в WSDL также представляется в XML-формате.

Архитектура Web-служб

В архитектуре Web-служб рассматриваются две ее составные части: базовая (Basic Architecture) и расширенная (Extended Web Services Architecture) архитектура. **Базовая архитектура** включает в себя стандартный (обязательный) набор средств этой технологии, уже нашедший применение в различных реализациях Web-служб; **расширенная архитектура** в большей части ориентирована на дополнительные или перспективные спецификации, расширяющие возможности технологии, такие как:

- асинхронные сообщения,
- передача данных в режиме приложений (attachment) SOAP-сообщений,
- идентификация и конфиденциальность сообщений и др.

Базовая архитектура

Базовая архитектура включает следующие технологии Web-служб:

- обмен сообщениями,
- обеспечение удаленного вызова процедур,
- описание услуг Web-служб,
- регистрация (публикация) и поиск (обнаружение) описаний услуг.

Базовая архитектура Web-служб определяет взаимодействие между агентами программного обеспечения как обмена сообщениями между *запрашивающими* сервис (requesters) и *поставщиками* услуг (providers - провайдерами). Requesters - агенты программного обеспечения, которые запрашивают выполнение сервиса, провайдеры - агенты программного обеспечения, которые обеспечивают выполнение сервиса. Каждый из агентов может быть одновременно и запрашивающим и провайдером. Провайдеры ответственны за реализацию сервиса и *публикацию* его описания, запрашивающие - должны иметь способ *поиска* описания услуг. Итак, базовая архитектура Web-служб проявляется в выполнении трех ролей: провайдера сервиса, агентства регистрации и поиска сервиса и клиента, запрашивающего сервис. Взаимодействие включает операции *декларации* (публикации), *поиска* и *связывания* (bind) сервиса. Эти роли и операции являются для Web-служб артефактами (artifacts): модулями программного обеспечения и их описаниями. В типичном сценарии Web-служб провайдер сервиса содержит программные модули с сетевым доступом, реализующие сервис, он же формирует описание сервиса и его декларацию (публикацию), доступную клиенту-потребителю или агентству регистрации (service discovery agency).

Клиент-потребитель выполняет операции поиска для получения описания сервиса локально или через регистрационное агентство и, используя это описание, выполняет процедуру *связывания* с провайдером сервиса, инициирует взаимодействие с модулями, реализующими данный сервис.

На рис.1 изображена схема взаимодействия провайдера сервиса, регистрационного агентства и клиента-потребителя сервиса.

Как выше уже отмечалось, основной единицей обмена в технологии Web-служб являются сообщения, имеющие структуру XML-документа. Описания сервиса используют также XML-нотацию, содержащую все необходимые детали для взаимодействия в сети, включая описание форматов сообщения, транспортных протоколов и его локализацию. Иначе говоря, формирование и транспортировка XML-сообщений подчиняются правилам и спецификациям SOAP-протокола; провайдер *публикует* WSDL-файл, содержащий описание семантики и синтаксиса сообщения, конечного адреса сервиса (endpoint), дающего возможность клиенту-потребителю правильно сгенерировать SOAP-сообщение и отправить его в правильном направлении. Сервис-провайдер получает это сообщение, обрабатывает содержащий в нем запрос и отправляет результат клиенту также в виде SOAP-сообщения. Технология, типичная для такого типа взаимодействия Web-служб, включает SOAP, WSDL и HTTP.

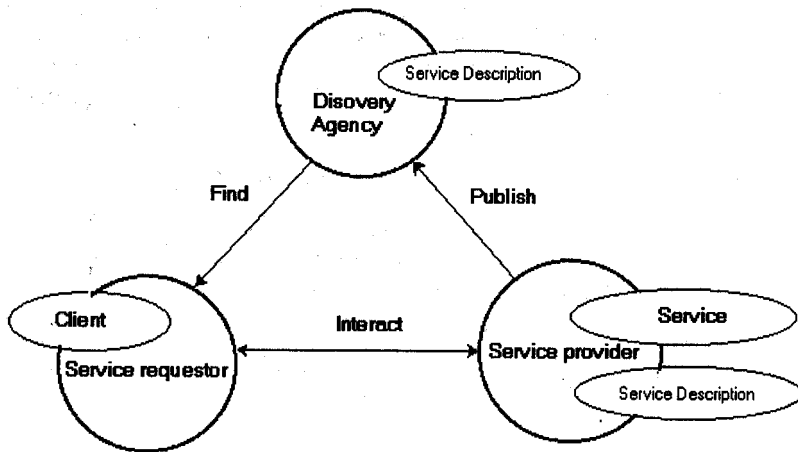


Рис. 1. Схема взаимодействия провайдера сервиса, регистрационного агентства и клиента-потребителя сервиса

SOAP, Simple Object Access Protocol [2,3]

SOAP - достаточно простой, основанный на XML-механизме, способ создания структурированных пакетов данных для обменов между сетевыми приложениями. SOAP содержит четыре основных компонента:

- *конверт* (envelope), определяющий рамочную структуру сообщения в формате XML,
- *набор правил* для представления типов данных,
- *соглашение* о представлении вызова удаленных процедур (в режиме RPC),
- *правила* совместного выполнения протоколов SOAP и HTTP. SOAP может использовать также комбинацию различных сетевых протоколов, таких как HTTP, SMTP, FTP, RMI/IIOP.

WSDL, Web Services Definition Language [4]

И отправитель, и приемник SOAP-сообщения должны иметь доступ к одному и тому же описанию сервиса. Отправителю нужно в описании сервиса, чтобы знать, как правильно форматировать сообщение, а приемник - для правильной его интерпретации. Файл WSDL формируется в формате XML-документа самим отправителем или провайдером сервиса и содержит всю необходимую для потенциальных потребителей сервиса информацию в формализованном виде. Иными словами, знание сетевого адреса WSDL-файла в Интернете является достаточной основой для понимания и правильного построения запросов от клиента-потребителя сервиса.

Определения Web-сервиса в WSDL могут *транслироваться* (отображаться) на многие языки, применяемые для реализаций Web-служб: Java, C++, Cobol и др.

Описание сервисов представляет собой XML-документ, состоящий из нескольких элементов, в том числе из описания *пространства имен* (namespace), описания типов и элементов сообщений, порта, а также возможных операций (для RPC) — запросов и ответов. WSDL-файл, содержащий описание сервисов, является достаточно сложным документом, поэтому для его создания и анализа, как правило, пользуются автоматическими генераторами, включенными в состав средств разработки.

Service discovery agency (регистрационное агентство)

Выше уже упоминалось об использовании не являющегося необходимым компонента Web-служб, так называемого регистрационного агентства или *репозитария* сервиса. В репозиторий направляется (публикуется) провайдером сервиса в том или ином формате (свободном или формализованном виде) информация об этих услугах. Клиент-потребитель услуг по различным критериям может обнаруживать и исследовать их качество и количество и, после выполнения процедуры *связывания* (binding), непосредственно обращаться к поставщику для выполнения декларированного сервиса. Для поиска и анализа сервиса *необязательно* иметь вышеописанную службу. Вся необходимая информация для подключения к сервису может передаваться провайдером потребителю конфиденциальным путем в виде записок, на дискетах, либо используя сетевые средства обмена данными типа FTP или WWW [5,6] и т. д. Консорциум W3C не занимается выработкой рекомендаций (стандартов) для информационного обслуживания поставщиков (провайдеров) сервисов. Для этого существуют уже зарекомендовавшие себя в электронном бизнесе системы типа ebXML или UDDI.

Отметим ряд реализаций регистрационных служб:

1. **UDDI** (Universal Description, Discovery and Integration) [7] представляет собой средство регистрации вычислительных или коммерческих услуг в формализованном XML-формате. UDDI можно рассматривать как логическую структуру, имеющую общепринятую стандартную классификацию:

- *желтые страницы*, в которых регистрируются под различными категориями деловые предложения и услуги.
- *белые страницы*, которые содержат контактную информацию.

- *зеленые страницы*, содержащие в основном технические детали для обращения к сервису.

UDDI предоставляет свои услуги также в формате Web-служб, и к нему можно обращаться за получением сервиса стандартными средствами, например SOAP-сообщениями. Ниже приводится фрагмент программы **pingUDDI.java**, с подготовкой SOAP-запроса с задаваемым ключевым словом в переменной **keyWord**:

```

.....
String keyWord = "Microsoft";
// Create the connection and the message factory
SOAPConnectionFactory scf = SOAPConnectionFactory.newInstance();
SOAPConnection connection = scf.createConnection();
MessageFactory msgFactory = MessageFactory.newInstance();

// Create a message
SOAPMessage msg = msgFactory.createMessage();

// Get the envelope from the message's SOAP part
SOAPEnvelope envelope = msg.getSOAPPart().getEnvelope();

// Get the SOAP body from the envelope
SOAPBody body = envelope.getBody();

SOAPBodyElement findBusiness = body.addBodyElement
    (envelope.createName("find_business", "", "urn:uddi-org:api"));
findBusiness.addAttribute(envelope.createName("generic"), "1.0");
findBusiness.addAttribute(envelope.createName("maxRows"), "100");
SOAPElement businessName = findBusiness.addChildElement
    (envelope.createName("name"));

businessName.addTextNode(keyWord);

URLEndpoint endpoint
    = new URLEndpoint(System.getProperties().getProperty("URL"));
msg.saveChanges();

SOAPMessage reply = connection.call(msg, endpoint);

System.out.println("Received reply from: " + endpoint);
reply.writeTo(System.out);
.....

```

Существует ряд открытых для пользователей UDDI-серверов, как для промышленного использования (UBR, UDDI Business Registry), так и для исследовательских целей (тестирования):

- SAP UDDI Test Business Registry (<http://udditest.sap.com/>);
- IBM UBR Node (<http://uddi.ibm.com/>);
- Microsoft UBR Node (<http://uddi.microsoft.com/>);

- Microsoft Test Node (<http://test.uddi.microsoft.com/>);
- и другие (<http://www.uddi.org/solutions.html>).

2. **ebXML** [8], в основном предназначенный для электронного бизнеса, также предоставляет средства регистрации, поиска и анализа сервисных услуг в форматах **CPP** (Collaboration-Protocol Profile) и **CPA** (Collaboration-Protocol Agreement) в XML-представлениях. В отличие от **UDDI**, где информация строго структурирована и формализована, т.е. содержит только *метаданные* о сервисе, **ebXML** допускает включение в регистрацию кроме метаданных и дополнительную информацию произвольной структуры. Простой пример: файл **WSDL**, столь необходимый для клиента-потребителя сервиса, не может включаться непосредственно в **UDDI**-описания (здесь возможна лишь ссылка на этот файл на стороне провайдера услуг), **ebXML** допускает включение в регистрацию любых технических деталей, касающихся декларируемого сервиса.

3. **JAXR** (Java API for XML Registries) [9] – является составной частью свободно распространяемого пакета **JWSDP** компании Sun, в котором реализована поддержка **Web-служб**. Надо подчеркнуть особенности этой службы регистрации:

- пользователь может самостоятельно устанавливать сервер регистрации **JAXR** и соответствующую базу данных;
- информационная модель **JAXR** (см. рис. 2) основана на **RIM**-модели регистрирующей системы **ebXML** (**RIM**, Registry Information Model) с расширенной поддержкой **UDDI**. Таким образом, **JAXR** является совместимым с обеими моделями **UDDI** и **ebXML**. Надо отметить, что в редакции 1.0 пакета **JWSDP** реализована только **UDDI**-поддержка.

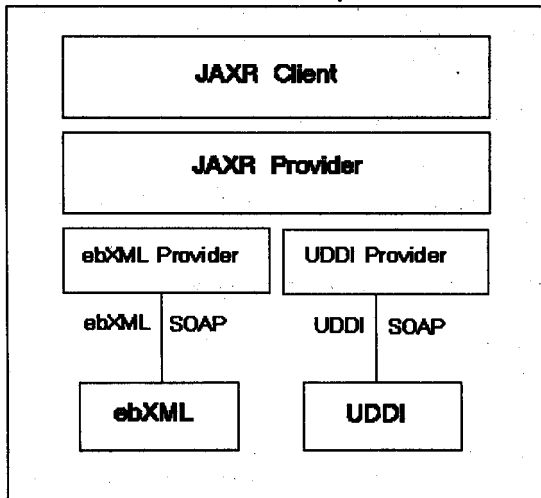


Рис. 2. Архитектура верхнего уровня JAXR

Структура SOAP-сообщений

Обмен данными между клиентом-потребителем и провайдером – поставщиком услуг – это передача SOAP-сообщений. Протокол SOAP предлагает способ формирования сообщений в XML-формате, основанный на объектной модели XML-документа DOM (Document Object Model). SOAP-сообщения – это иерархическая структура вложенных XML-элементов (или узлов): SOAPMessage, SOAPPart, SOAPEnvelope, SOAPHeader, SOAPBody и содержимое передаваемого сообщения, также в формате XML (XML content). На рис. 3 представлена иерархическая структура SOAP-сообщения.

Типы взаимодействия в Web-службах

Разработчики серверной и клиентской частей Web-служб могут использовать два типа предоставления и потребления вычислительных услуг, внешне принципиально разных:

- передача данных (запрос и ответ) в формате SOAP-сообщений. Причем формирование XML-структуры сообщения, его кодирование и обработка результата выполняются “вручную” прикладными программистами с использованием объектной модели DOM и SOAP-протокола.
- использование механизма вызова удаленных процедур RPC (Remote Procedure Call) для передачи данных и получения результатов в обращениях к подпрограммам серверного объекта. Здесь важную роль играет умение пользоваться описаниями интерфейсов удаленного объекта в WSDL.

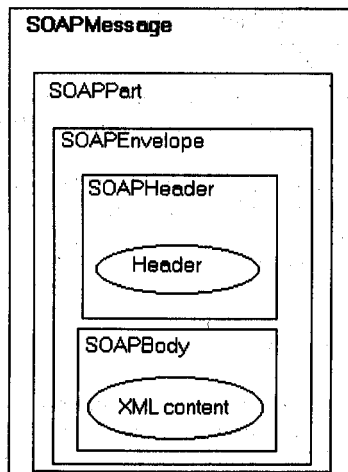


Рис. 3. Иерархическая структура SOAP-сообщения

Передача SOAP-сообщений

Сообщения могут передаваться в *синхронном* или *асинхронном* режиме. В синхронном режиме (request-response messaging) передающий процесс блокируется на время ожидания ответа, который также должен соответствовать SOAP-стандарту. В асинхронном режиме (one-way messaging) для передач сообщений используются посредники (провайдеры). На время обменов передающий процесс не блокируется, но время возврата результата – непредсказуемо. Управление типами сообщений выполняется выбором соединения клиента и провайдера (объектов SOAPConnection или ProviderConnection). Для передачи синхронных сообщений используется метод call() объекта SOAPConnection, параметром которого является структура, представляющая объектную модель SOAP-сообщения, а результатом выполнения запроса – также SOAP-сообщение в формате DOM. Для передачи асинхронных запросов используется объект ProviderConnection и его метод send().

В приведенном выше фрагменте программы для обращения к UDDI-серверу показана типичная последовательность выполнения SOAP-запроса, вот некоторые из операций:

Выполнение соединения:

```
SOAPConnectionFactory scf = SOAPConnectionFactory.newInstance();  
SOAPConnection connection = scf.createConnection();
```

Создание объекта для SOAP-сообщения:

```
SOAPMessage msg = msgFactory.createMessage();  
SOAPEnvelope envelope = msg.getSOAPPart().getEnvelope();
```

Создание объекта “элемент” для размещения содержимого запроса

```
SOAPBody body = envelope.getBody();
```

Затем выполняется последовательность операторов для создания запроса в виде XML-объекта, который включается в элемент типа SOAPBody.

Обращение к методу call для передачи SOAP-сообщения msg

```
SOAPMessage reply = connection.call(msg, endpoint);
```

Распечатка принятого результата

```
reply.writeTo(System.out);
```

Удаленный вызов процедур, RPC

Режим RPC во многом напоминает технологии обмена данными, используемые в объектных распределенных системах типа RMI и OMG CORBA. Основные моменты развертывания RPC:

- разработка интерфейса удаленного сервисного объекта (формальное описание методов объекта, типа параметров и результата), реализующего вычислительный сервис (Web Service);
- кодирование методов сервисного объекта (implementation);

- создание описания методов объекта в WSDL-документе;
- декларация WSDL-документа;
- обработка (поиск и анализ) клиентской программой WSDL-документа;
- связывание (binding) - подключение к серверу (провайдеру сервиса);
- формирование объектной ссылки для методов сервисного объекта;
- обращение к методам сервиса и получение результатов вычислений.

Надо отметить два существенных момента использования RPC:

- поскольку единственным форматом передач данных в технологии Web-служб являются SOAP-сообщения, серьезной проблемой является конвертирование всех вызовов процедур и их данных в стандартный формат, принятый в SOAP, т.е. в XML-структуру. Это же обстоятельство и позволяет обеспечить реальную совместимость всех участвующих во взаимодействиях компонентов распределенной системы, независимо от применяемых поставщиками услуг и/или клиентами-потребителями языков программирования, операционных систем и т.д.
- взаимное конвертирование SOAP-сообщений и RPC-запросов выполняют специальные программы – представители противоположной стороны или *суррогаты*. На стороне клиента они называются стабами (**stubs**), а в серверной программе для Web-служб они называются **tie**. Эти программы-посредники генерируются автоматически при подготовке и развертывании клиентских и серверных компонентов Web-служб. Уровень знания их и применения прикладным программистом зависит от конкретной программной реализации Web-службы компанией-производителем.

На рис. 4 представлена диаграмма прохождения и конвертирования RPC-запросов.

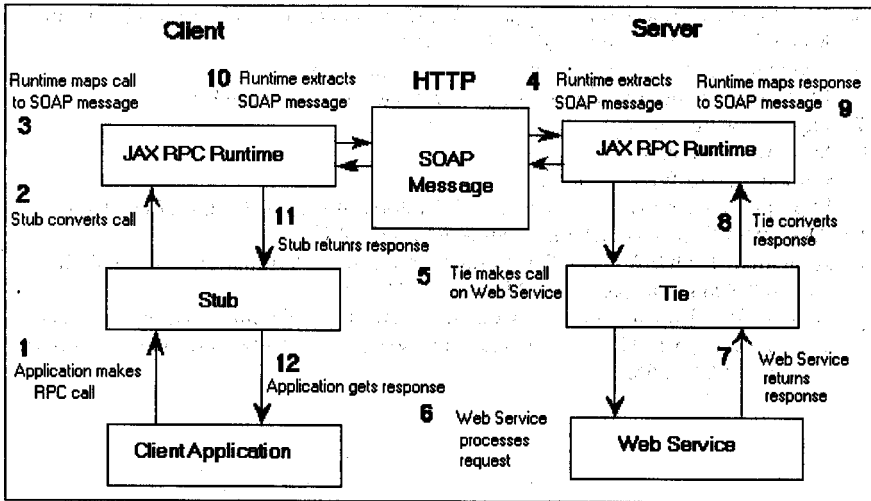


Рис. 4. Диаграмма преобразований RPC-вызовов в SOAP-сообщения

Заключение

Web Services – относительно новая технология для распределенных вычислительных сред. Основная ее особенность – это объединение в рамках WWW-консорциума многих крупных, иногда антагонистических, фирм-производителей, для разрешения важнейшей и противоречивой проблемы совместимости различных конкурирующих технологий интеграции неоднородных компьютерных систем, использующих одну и ту же идеологию объектной ориентации. Основой обеспечения такой совместимости является последовательное применение на всех уровнях предложенной технологии XML-представления структур данных, форматов передаваемых сообщений, интерфейсов взаимодействия распределенных компонентов. С другой стороны, основные компоненты и протоколы Web Services – это уже зарекомендовавшие себя в других областях применения технологии: SOAP, WSDL, UDDI, eXML и HTTP/SMTP. В немалой степени это сближение позиций фирм-производителей является следствием давления рынка электронного бизнеса.

Надо отметить также и то обстоятельство, что существует стремление использовать предложенную совместимость Web Services и для решения так называемых “решеточных” задач или GRID-технологий [11], получивших, в частности, широкое применение в задачах физики высоких энергий [12].

Литература

1. W3C Consortium. <http://www.w3c.org/>.
2. SOAP Version 1.2 Part 1: Messaging Framework. <http://www.w3c.org/TR/soap12-part1/>.
3. SOAP Version 1.2 Part 2: Adjuncts. <http://www.w3c.org/TR/soap12-part2/>.
4. Web Services Description Language (WSDL) Version 1.2. W3C Working Draft 3 March 2003. <http://www.w3.org/TR/wsdl12>.
5. XMethods Utility Services. <http://www.xmethods.net/>.
6. DS Data Systems and Web Services. <http://www.dsdata.co.uk/WS/>.
7. <http://www.uddi.org/>.
8. <http://www.ebxml.org>.
9. Introduction to Web Services. <http://java.sun.com/webservices/docs/1.1/tutorial/doc/IntroWS.html#wp70565>.
10. Document Object Model. <http://www.w3.org/DOM/>.
11. Доменико Т. OGSA: где GRID встречается с Web, “Открытые системы”, № 1, изд-во “Открытые системы”, Москва, 2003, с. 47.
12. Ильин В., Кореньков В., Солдатов А. Российский сегмент глобальной инфраструктуры LCG, “Открытые системы”, № 1, изд-во “Открытые системы”, Москва, 2003, с. 56.

Получено 18 июля 2003 г.

**Издательский отдел
Объединенного института ядерных исследований
предлагает Вам приобрести перечисленные ниже книги:**

Индекс книги	Название книги
E2-2000-226	Труды V международного рабочего совещания «Физика тяжелых кварков». Дубна, 2000, 154 с. (на англ. яз.)
Д9-2000-238	Труды международной школы молодых ученых «Проблемы ускорения заряженных частиц». Дубна, 1999, 240 с. (на русском и англ. яз.)
E1,2-2000-244	Труды международного совещания «Физика больших множественностей». Дубна, 2000, 204 с. (на англ. яз.)
E2-2000-248	Труды II международного симпозиума «Физика и детекторы на LHC». Дубна, 2000, 3 тома (на англ. яз.)
Д1-2000-279	Труды XVII международного совещания «Коллаборация EMU 01 и перспективы использования фотозмульсий в экспериментах на пучках релятивистских ядер нуклотрона». Дубна, 1999, 244 с. (на русском и англ. яз.)
E1,2-2000-282	Труды международного совещания «Актуальные проблемы астрофизики». Дубна, 2000, 406 с. (на англ. яз.)
Д19-2001-5	Труды международной конференции «Современные проблемы радиобиологии и эволюции». Дубна, 2000, 493 с. (на русском и англ. яз.)
E10-2001-11	Труды 2 международного совещания «Системы сбора данных в экспериментах на нейтронных источниках». Дубна, 2000, 341 с. (на англ. яз.)
E1,2-2001-76	Труды международного совещания «Релятивистская ядерная физика от сотен МэВ до ТэВ». Словакия, Стара Лесна, 2000, 328 с. (на англ. яз.)
E3-2001-192	Труды IX Международного семинара по взаимодействию нейтронов с ядрами. Дубна, 2001, 500 с. (на англ. яз.)
E14-2001-215	Труды II германо-российского совещания пользователей реактора ИБР-2. Нейтронные исследования в области физики конденсированного состояния на реакторе ИБР-2. Дубна, 2001, 102 с. (на англ. яз.)
Д9-2001-271	Труды II международного рабочего совещания «Синхротронный источник ОИЯИ: перспективы исследований». Дубна, 2001, 157 с. (на русском и англ. яз.)
E5,11-2001-279	Труды международного совещания «Компьютерная алгебра и ее приложения в физике». Дубна, 2001, 359 с. (на англ. яз.)
E1,2-2001-290	Труды международного совещания «Релятивистская ядерная физика: от сотен МэВ до ТэВ». Дубна, 2001, 2 тома: 300 с. и 275 с. (на англ. яз.)
E1,2-2001-291	Труды XV международного семинара по проблемам физики высоких энергий. «Релятивистская ядерная физика и квантовая хромодинамика». Дубна, 2000, 2 тома: 391 с. и 382 с. (на англ. яз.)

- Д9-2002-23 Труды IV научного семинара памяти В. П. Саранцева. Дубна, 2001, 263 с. (на русском и англ. яз.)
- Д10,11,-2002-28 Труды XVIII Международного симпозиума по ядерной электронике и компьютерингу (NEC'2001). Болгария, Варна, 2001, 261 с. (на русском и англ. яз.)
- Е1,2-2002-45 Труды совещания «Физика очень больших множественностей». Дубна, 2001, 191 с. (на англ. яз.)
- Е2-2002-48 Труды XVI международного совещания «Суперсимметрии и квантовые симметрии». Польша, 2001, 276 с. (на англ. яз.)
- Е4-2002-66 Труды семинара «Перспективы в изучении структуры ядра и ядерных реакций». Дубна, 2002, 112 с. (на англ. яз.)
- Е2-2002-70 Труды XXIII Международного коллоквиума по теоретико-групповым методам в физике. Дубна, 2000, 2 тома: 667 с. (на англ. яз.)
- Е15-2002-84 Труды V международного рабочего совещания «Применение лазеров в исследованиях ядер. Перспективы развития лазерных методов исследования ядерной материи». Познань, Польша, 2001, 353 с. (на англ. яз.)
- Е18-2002-88 Труды международной летней школы «Ядерно-физические методы и ускорители в биологии и медицине». Дубна, 2001, 221 с. (на англ. яз.)
- Д19-2002-95 Труды II международного симпозиума и II Сисакьяновские чтения «Проблемы биохимии, радиационной и космической биологии». Дубна, 2001, 2 тома: 249 с. (на русском и англ. яз.)
- Е2-2002-103 Труды IX рабочего совещания по физике спина при высоких энергиях. Дубна, 2001, 389 с. (на англ. яз.)
- Е7,17-2002-135 Труды VI рабочего совещания «Теория нуклеаций и ее применение». Дубна, 2000–2002, 513 с. (на англ. яз.)
- Е1,2-2002-166 Труды международной школы-семинара «Актуальные проблемы физики частиц». Гомель, 2001, 2 тома: 328 с. и 292 с. (на англ. яз.)
-

За дополнительной информацией просим обращаться в издательский отдел ОИЯИ по адресу:

141980, г. Дубна, Московская обл.,
ул. Жолио-Кюри, 6.
Объединенный институт ядерных исследований,
издательский отдел.
E-mail: publish@pds.jinr.dubna.su

Галактионов В. В.

P10-2003-140

Web Services — сервис-ориентированная технология для распределенных объектных вычислительных систем. Основные концепции, протоколы и спецификации

Приведены результаты исследования новой сервис-ориентированной технологии Web Services для распределенных объектных вычислительных систем. Также по официальным материалам WWW-консорциума (W3C) описаны основные концепции, протоколы и спецификации этой технологии. Эта технология в ближайшее время будет являться основой для решения так называемых «решеточных» задач или GRID-технологий, получивших, в частности, широкое применение в задачах физики высоких энергий.

Работа выполнена в Лаборатории информационных технологий ОИЯИ.

Сообщение Объединенного института ядерных исследований. Дубна, 2003

Перевод автора

Galaktionov V. V.

P10-2003-140

Web Services — Service-Oriented Technology for the Distributed Object Computing Systems. The Basic Concepts, Protocols and Specifications

The results of research of new technology Web Services for the distributed object-oriented computing systems are given. On official materials of a WWW consortium (W3C) the basic concepts, protocols and specifications of this technology are also described. This technology in the near future will be a basis for solving the tasks for GRID-technologies which have received, in particular, wide application in tasks of high energy physics.

The investigation has been performed at the Laboratory of Information Technologies, JINR.

Communication of the Joint Institute for Nuclear Research. Dubna, 2003

Редактор *М. И. Зарубина*
Макет *Н. А. Киселевой*

Подписано в печать 16.09.2003.

Формат 60 × 90/16. Бумага офсетная. Печать офсетная.

Усл. печ. л. 0,62. Уч.-изд. л. 0,88. Тираж 300 экз. Заказ № 54102.

Издательский отдел Объединенного института ядерных исследований
141980, г. Дубна, Московская обл., ул. Жолио-Кюри, 6.

E-mail: publish@pds.jinr.ru

www.jinr.ru/publish/