

Служба управления потоками работ по манипулированию ресурсами репозитория

Нестеренко А.К., *Бездушный А.А.**, *Сысоев Т.М.***, *Бездушный А.Н.***,
*Серебряков В.А.*****

***МФТИ, **ВЦ РАН**

В работе рассматриваются проблемы и задачи, возникающие при манипулировании ресурсами репозитория цифровых библиотек, информационных Интернет систем и порталов. Анализируются соответствующие требования к службе управления потоками работ, которая в этом случае должна обеспечить управление созданием и модификацией информационных ресурсов произвольных типов. Описывается реализация такой службы, ее роль и место в новой версии системы ИСИР [8, 9, 10]. Реализация следует стандартам WfMC [4]; канонической модели и языку спецификации потоков работ [5]. Решение демонстрируется в применении к управлению потоками работ по сопровождению ресурсов Информационного Web-портала РАН [11].

1 Введение

За последние 15 лет были разработаны средства, позволяющие не только выполнять конкретные работы, но и управлять их потоками (Workflow). Эти процессы формально специфицируются в компьютерных системах. Потоки работ управляются компьютерной программой, которая назначает задания, принимает их и фиксирует степень их исполнения. Традиционно понятие «рабочий процесс» определяется с позиций офисных работ, работ с документами – передача документа, подписание приказа, подготовка накладной. Но те же самые принципы и понятия встречаются, например, в работе промышленного предприятия – подготовка документов, деталей, станков и рабочего персонала для сборки сложных технологических систем.

С появлением сложных Web-систем, цифровых библиотек, корпоративных порталов, поддерживающих и управляющих большими объемами и потоками информации, сформировалась категория подсистем, называемых системами управления содержанием/контентом Web-систем (WCMS - WebContentManagementSystem). Первые WCMS представляли совокупности Web-форм для управления данными – как несвязанных Web-форм, так взаимосвязанных web-форм, но с фиксированным в программном коде порядком обработки. Сейчас актуальной становится потребность в более гибкой организации управления потоками работ, в поддержке декларативных форм определения и оперативного

изменения потоков работ. WCMS должны управлять созданием и модификацией информационных, сильно взаимосвязанных ресурсов произвольных типов. Потоки работ могут быть связаны не только с поддержкой информационных ресурсов, но и с выполнением полностью автономных регламентных процедур системы в ответ на, например, программные обращения, на наступление некоторого события и т.п. Описанию попытки создания такой системы, требованиям к ней, проблемам реализации и применению посвящена эта работа.

2 Требования к функциональности системы управления потоками работ

Системы управления потоками работ обеспечивают процедурную автоматизацию делового процесса, управляют последовательностью действий работника с ресурсами, связанными с различными шагами деятельности. Индивидуальный деловой процесс может иметь время жизни в пределах от нескольких минут до нескольких дней (или даже месяцев или лет) в зависимости от его сложности и продолжительности различных действий. На самом высоком уровне все workflow системы могут быть характеризованы как системы, обеспечивающие поддержку в трех функциональных областях:

- Встроенные функции, которые задаются определением и моделированием процесса workflow (Build-timefunctions).
- Функции контроля во время выполнения, отвечающие за управлением процессами workflow в эксплуатационной среде и последовательностью различных действий, которые будут обработаны как часть каждого процесса (Run-timecontrolfunctions).
- Взаимодействие во время выполнения с пользователями или программными системами (Run-timeActivityInteractions).

Следующая диаграмма иллюстрирует указанные выше основные характеристики workflow-систем и отношений между этими основными функциями:



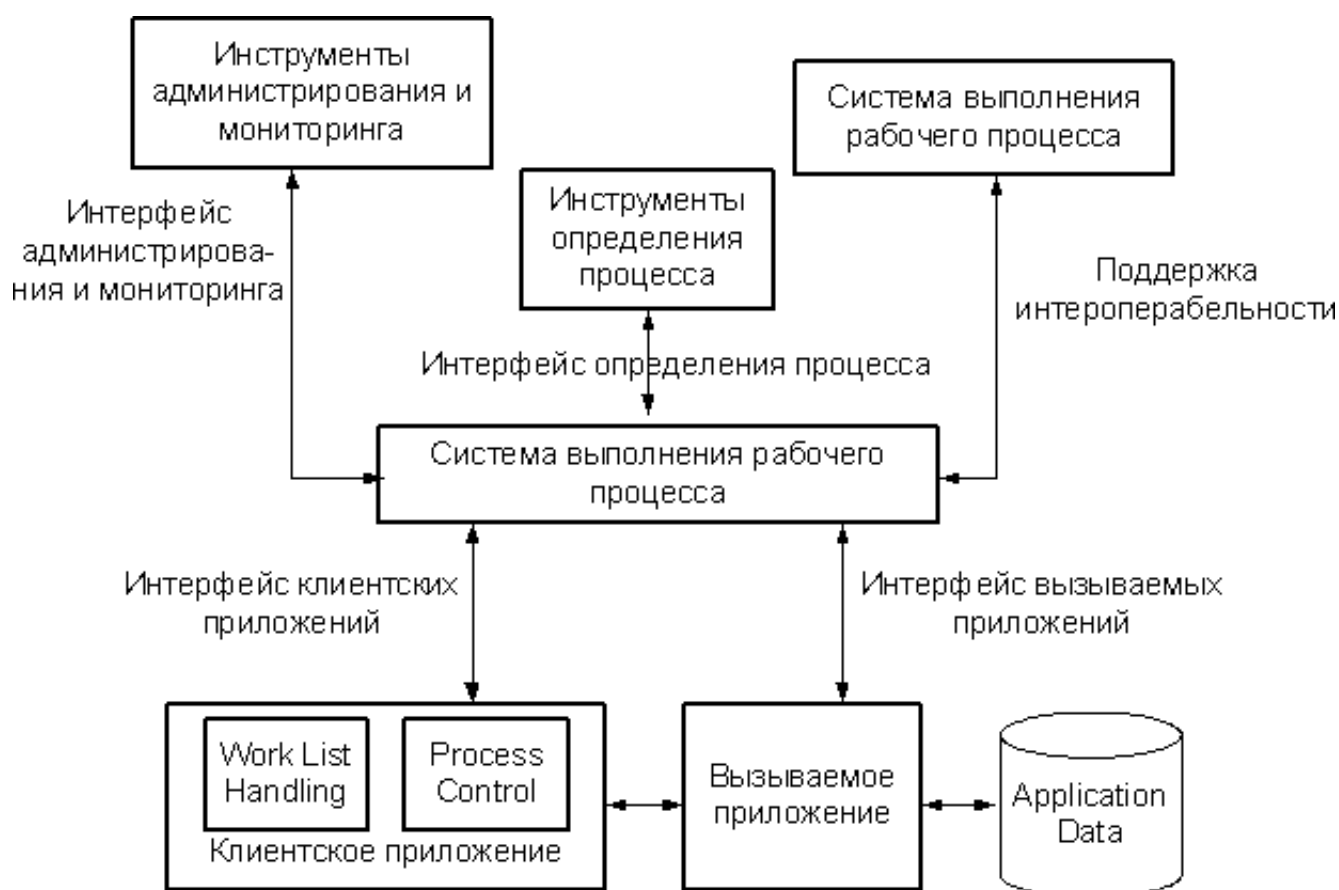
Рассмотрим назначение указанных архитектурных уровней системы управления потоками работ:

- *Уровень определения процесса (Built-timeFunctions).* Данный уровень представлен набором функций, задача которых заключается в декларативном определении бизнес-процесса. Во время этой фазы бизнес-процесс переводится из «реального» мира в «формальный», понимаемый компьютером, с помощью анализа, моделирования и использования набора средств декларативного представления. Результирующая декларация процесса носит название «модели процесса», «шаблона процесса», «метаданных процесса» или «определения процесса». Модель процесса состоит из ряда дискретных «действий», переход между которыми осуществляется в соответствии с набором некоторых правил посредством выполнения операций, исполнителем которых может быть как человек, так и компьютер. Определение процесса может быть представлено как текстовым или графическим описанием, так и декларацией на некотором формальном языке описания потоков работ. Некоторые процессы допускают возможность внесения изменений в их определение в ходе их выполнения. Соответствующая функциональность помечена обратной стрелкой на диаграмме.
- *Уровень создания и выполнения процесса (Run-timefunctions).* На этапе выполнения происходит анализ определения процесса специальным программным обеспечением, которое отвечает за создание и управление экземплярами потоков работ, управление и планирование заданий,

привлечение соответствующих человеческих, IT- и других видов ресурсов. Функции уровня выполнения являются связующим звеном между моделью процесса в формальном определении и его реальным видом, отражающим взаимодействие пользователей и IT-приложений. Центральным компонентом системы является служба управления потоками работ, которая отвечает за создание и удаление процессов, планирование заданий процесса и взаимодействие с человеческими и автоматизированными ресурсами.

- *Уровень средств взаимодействия с пользователями и приложениями (RuntimeActivityInteractions).* Индивидуальные задания рабочего процесса, как правило, связаны с человеческой деятельностью, для выполнения которой привлекаются дополнительные IT-приложения (например, ряд форм для заполнения). Некоторые операции требуют наличия специальных средств для манипулирования данными (например, занесение новой записи в базу данных). Поэтому необходимо обеспечить гибкое взаимодействие пользователей со службой управления потоками работ для того, чтобы передавать управление между заданиями, назначать операционные состояния процессам, запускать приложения или передавать необходимые данные.

Следующая диаграмма отражает набор требований к функциональности workflow-систем [4]:



Выделяют следующие функциональные области, адресованные Workflow системам:

- Определение процесса работ: спецификация данных для процесса и его обмена с окружающей средой.
- Способность к взаимодействию: поддержание интероперабельности между различными системами workflow.
- Вызов сторонних приложений.
- Клиентские приложения: поддержание способности к взаимодействию с пользовательскими функциями интерфейса.
- Администрирование и контроль: поддержание интерфейсов для обеспечения контроля системы, а также для облегчения управления взаимодействием с окружающей средой.

3 Обзор и сравнение существующих подходов к описанию бизнес-процессов

Одним из наиболее важных требований к системам управления потоками работ является следование ряду стандартов. На данный момент на рынке программных продуктов сложилась непростая ситуация [2] со стандартизацией процесса построения систем управления потоками работ. Пока стандартизация workflow-систем носит больше теоретический, чем практический характер.

Между тем, следование каноническим спецификациям и стандартам дает ряд очевидных преимуществ как в случае систем управления потоками работ, так и в случае других практических решений:

- Строгая формализация процесса описания потоков работ: можно реализовать стандартный (возможно частично автоматизированный) процесс по анализу и построению новых моделей реальных рабочих процессов.
- Переносимость и интероперабельность: модели процессов, созданные в рамках одной системы, могут частично или полностью работать под управлением другой (возможно, удаленной) системы.
- Универсальность: расширяемость стандартов делает возможным применение единого механизма описания управления потоками работ в различных сферах деятельности.

Таким образом, следование ряду стандартов позволяет создавать автономные, открытые workflow системы, что, безусловно, расширяет область и способы их применения для моделирования различных рабочих процессов.

На данный момент различными разработчиками предложен ряд языков для описания рабочих процессов. К основным стандартам можно отнести следующие:

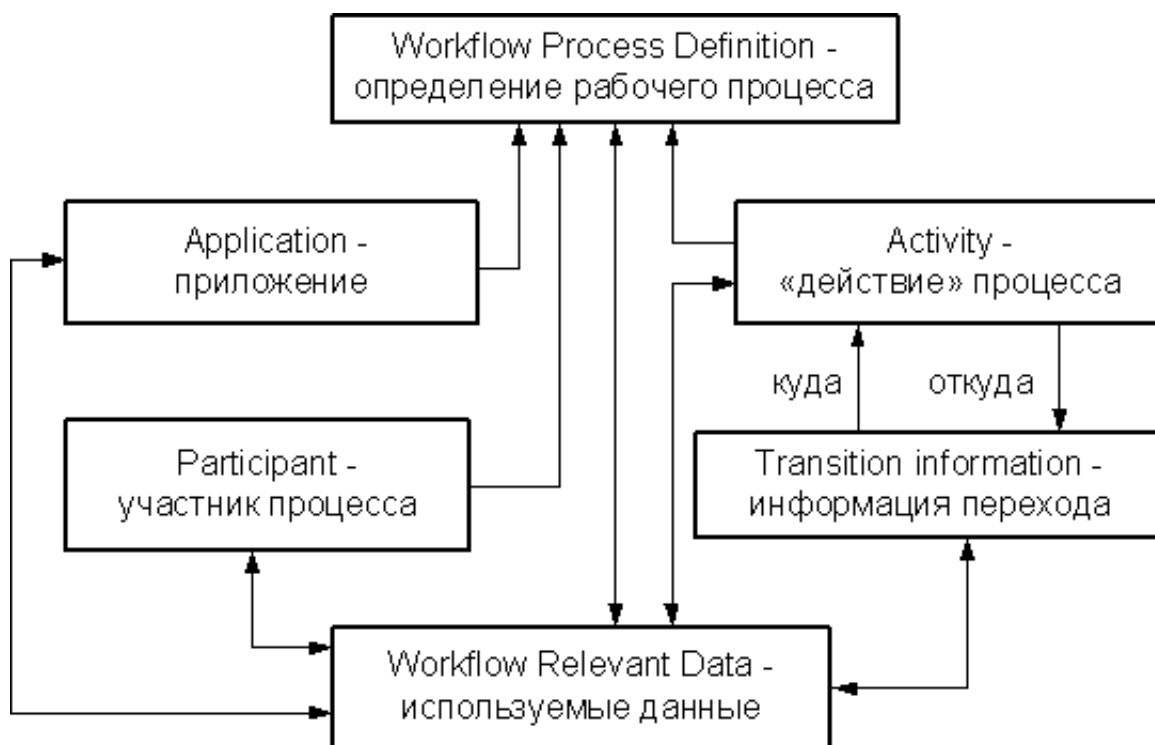
- XPDЛ (Workflow Management Coalition) [5];
- BPML (Business Process Management Initiative) [1];
- WSFL (IBM) [6];
- XLANG (Microsoft) [7].

Все перечисленные спецификации имеют свои характерные особенности, определяющие преимущества в функциональности основанного на их базе

решения.

3.1 XMLProcessDefinitionLanguage (XPDL)

Спецификация XPDL [4,5], предложенная WorkflowManagementCoalition, представляет собой формальную модель для описания рабочих процессов, относящихся к любым сферам деятельности. В соответствии с ней каждый поток работ разбивается на следующий набор взаимодействующих между собой компонентов:



- *WorkflowProcessDefinition* - представляет собой контекст выполняющегося процесса, и его данные могут быть доступны всем остальным компонентам.
- *Activity* - «действие» или «задание» процесса, представляющее собой этап, на котором происходит изменение содержания объектов процесса.
- *Transitioninformation* - переходы между заданиями (могут быть условными и безусловными).
- *WorkflowRelevantData* - оперативные данные, доступные всем компонентам процесса в ходе его выполнения.
- *Participant* - участник процесса, производящий «действия» над объектами и осуществляющий переходы (участники могут являться как человеческими, так и машинными ресурсами).
- *Application* - внешнее IT- или другое приложение, используемое для выполнения «действий».

Стандарт WfMC [4] определяет три фундаментальных понятия:

- *Workflow* - это процесс, определяемый как автоматизация некоторого делового процесса. Он включает в себя логику процесса и правила маршрутизации.
- *WorkflowManagement* - это набор методов и технологий для автоматизации

бизнес процесса в целом или его части посредством информационных объектов.

- *WorkflowManagementSystem (WfMS)* - это система, которая определяет, создает и управляет выполнением workflow-процессов с помощью специального программного обеспечения.

К характерным особенностям XPDЛ можно отнести следующее:

- С позиций описания XPDЛ рабочий процесс представляет собой направленный граф, узлами которого являются «действия», связанные между собой переходами. Переходы могут быть условными, причем условие проверяется на этапе выполнения конкретного «действия».
- В языке XPDЛ существует возможность выделения «блоков» – возможность объединения «действий» в блок «действий» со своими отдельными условными или безусловными точками входа и выхода. Так же имеется возможность определять вложенные подпроцессы внутри родительского процесса, которые сами по себе представляют полноценные потоки работ.
- Спецификация поддерживает возможность экспорта некоторых блоков описания одного процесса в описание другого с возможностью переопределения части импортируемого описания, что исключает необходимость дублирования идентичных фрагментов описания в нескольких процессах.
- XPDЛ является расширяемым стандартом. Он позволяет определять набор элементов и атрибутов, специфичных для конкретной сферы его применения.
- Элементы описания процессов XPDЛ имеют обширный набор атрибутов, определяющих ход выполнения процесса. К ним можно отнести условные выражения для переходов, временные рамки, задание множественных исполнителей «действий» и т.д.

3.2 The Business Process Modeling Language (BPML) и XLANG

Спецификации BPML и XLANG [1,7], предложенные BusinessProcessManagementInitiative и корпорацией Microsoft соответственно, являются родственными стандартами, реализующими блочную модель потоков работ. Рассмотрим спецификации данного семейства на примере BPML ввиду того, что данный стандарт охватывает наиболее широкий спектр характерных особенностей моделей рабочих процессов данного типа.

BusinessProcessModelingLanguage представляет собой абстрактную модель и XML-язык для описания рабочих процессов различного типа. В качестве исполнителей заданий потока работ в модели BPML выступают различного рода Web-сервисы. Для описания сервиса и всех услуг, предоставляемых им потребителю, используется спецификация WSDL (WebServicesDescriptionLanguage). Общение с такими службами может осуществляться, например, посредством протокола SOAP (SimpleObjectAccessProtocol).

В отличие от спецификации XPDЛ, стандарт BPML представляет собой блочную структуру. Элементарными блоками, из которых строится модель рабочего

процесса в BPML, являются «действия» (*Activity*). С позиций подобного блочного подхода сам рабочий процесс (в отличие от XPDЛ) не выделяется в виде отдельного объекта, а является просто специальным случаем агрегированного действия. В спецификации BPML выделяется три специальных типа процессов:

- Вложенный процесс (*NestedProcess*). Данный тип процесса, как и любое «действие», может быть включен в иерархию действий родительского (или на верхнем уровне корневого) процесса. При этом он образует свой собственный контекст.
- Процесс, обрабатывающий исключительные события (*ExceptionProcess*). Процесс этого типа, будучи указан в контексте родительского процесса, выполняет обработку системных событий, объявленных как исключительные.
- Процесс-компенсатор (*CompensationProcess*). После завершения работы любого процесса (с ошибкой или нет) процесс данного типа позволяет проводить действия по освобождению системных ресурсов и другие завершающие шаги.

Для общения между процессами (или «действиями») в спецификации BPML используются *сообщения* (*Messages*) и *сигналы* (*Signals*). Для этого процессы (или агрегированные «действия») должны назначать обработчиков сообщений или сигналов, а «действие»-источник должно генерировать сигналы (сообщения).

Двумя важными компонентами модели BPML являются *системные ошибки* (*Faults*) и *расписания* (*Schedules*). Системные ошибки (как и исключительные события) приводят к завершению выполнения текущего задания, в результате чего управление передается специальным обработчикам (системным или определенным в контексте). Расписания выполняют управляющие функции: в их задачу входит генерация специальных сообщений в определенные моменты времени, что позволяет осуществлять планирование запуска процессов.

Аналогом *WorkflowRelevantData* в XPDЛ (хранилище оперативных данных) в случае BPML является иерархия вложенных контекстов рабочих процессов.

3.3 The Web Services Flow Language (WSFL)

TheWebServicesFlowLanguage [6], разработанный компанией IBM, представляет собой XML-язык, описывающий композицию произвольного типа Web-служб в рамках одной модели потоков (*FlowModel*). Данная композиция описывается последовательностью точек доступа к функциям, предоставляемым различными службами. Порядок запуска сервисов определяется с помощью управляющих потоков и потоков данных между Web-службами.

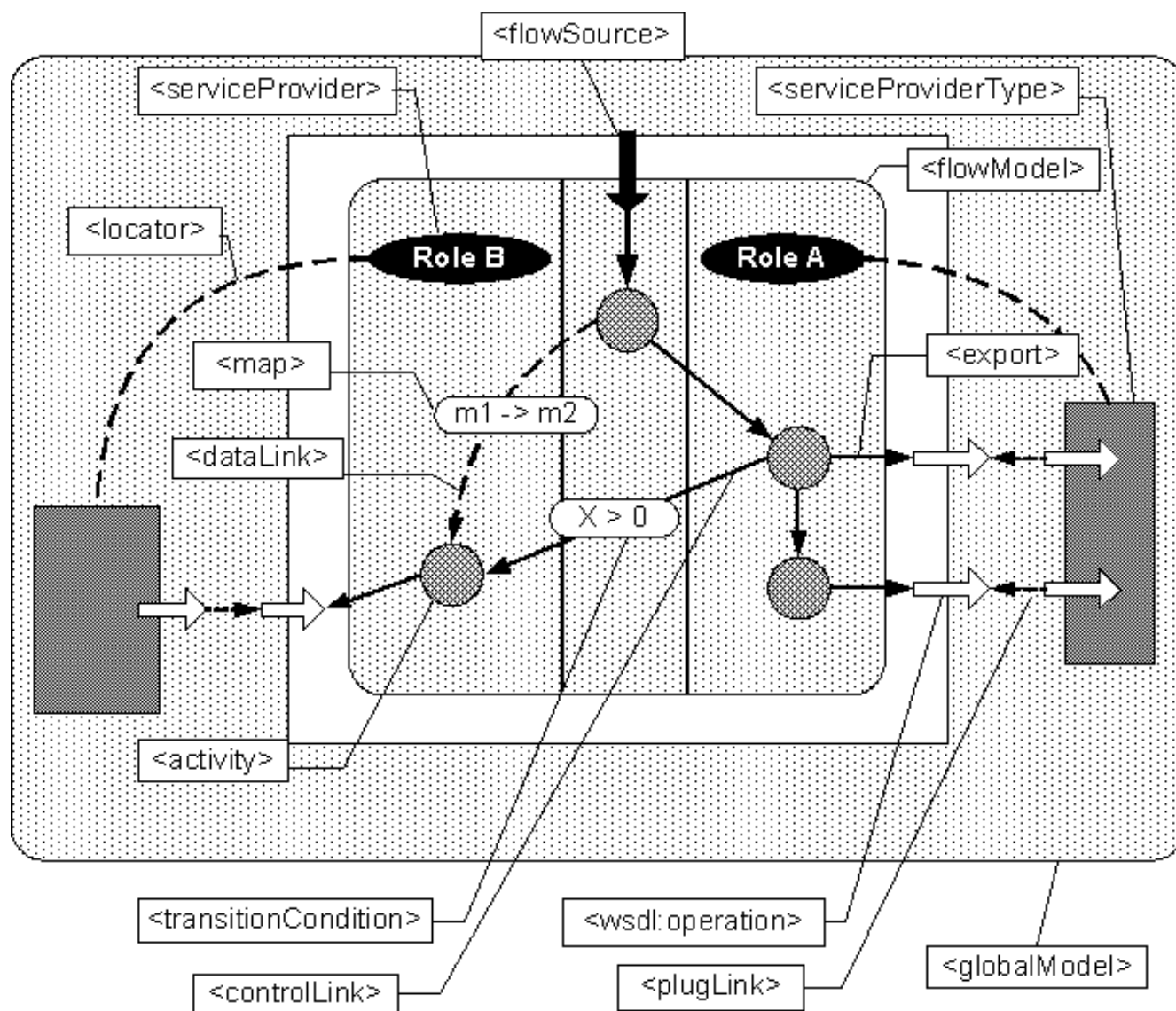
Каждый поток работ, объединяющий в себе использование набора некоторых Web-служб, включает в себя:

- Бизнес процесс, определяющий набор действий, которые необходимо выполнить в течение данного процесса.
- Управляющие правила, которые определяют последовательность

выполнения шагов бизнес-процесса.

- Поток информации между различными этапами бизнес-процесса.

Формальная WSFL-модель, отражающая взаимодействие между потоком работ и внешними поставщиками услуг, представлена на следующей диаграмме:



WSFLFlowModel определяет структуру бизнес-процесса: «действия» (помечены кружками на диаграмме) описывают шаги процесса, а «управляющие связи» (элемент **<controlLink>**) и «потоки данных» (элемент **<dataLink>**) определяют последовательность переходов и потоки данных между «действиями» соответственно. Для каждого «действия» назначен «поставщик услуг» (элемент **<serviceProvider>**), ответственный за выполнение данного этапа бизнес-процесса, и определяются связи между «действиями» в модели потоков и «операциями» (элемент **<wsdl:operation>**), предоставляемыми поставщиками услуг (элементы **<export>** и **<plugLink>**). Результирующая модель потоков работ (элемент **<flowModel>**) изображена в центре диаграммы. Она разбита вертикальными линиями на части, отражая тем самым связь «действий» процесса с конкретными поставщиками услуг.

Приведем список характерных особенностей спецификации WSFL:

- Этапы бизнес-процесса представлены «действиями», которые связываются в направленный граф с условными или безусловными переходами между его вершинами. Каждое «действие» связано с некоторой операцией провайдера услуг, отвечающего за его выполнение.
- Связи между «действиями» (ControlLink) представляют собой направленные дуги графа маршрута модели потоков. Переходы между «действиями» могут быть условными.
- Модель WSFL допускает возможность параллельного выполнения нескольких «действий». Для этого служит «действие» специального типа, называемое *forkactivity*, переход с которого может осуществляться одновременно на несколько других действий. При этом WSFL поддерживает набор средств для синхронизации одновременно выполняемых этапов процесса.
- Существует два выделенных типа «действий»: начальные (на которые не осуществляются переходы посредством использования ControlLink) и конечные (которые не связаны «исходящими» переходами с другими «действиями» процесса).
- Между последовательно выполняющимися «действиями» может быть организован поток данных для обмена управляющей и другой информацией.

Языки WSFL и XPDЛ относятся к одному семейству графовых XML-языков описания потоков работ, в отличие от блочной структуры BPML, что определяет их схожую функциональность, хотя XPDЛ и дает гораздо больший уровень абстракции (возможно использование человеческих ресурсов и не конкретизируются виды и интерфейсы автоматизированных).

3.4 Выбор стандарта для реализации службы управления потоками работ ИСИР

Как видно из приведенного выше обзора существующих на данный момент абстрактных моделей для описания реальных потоков работ все они разбиваются на две категории:

- Графовые модели (XPDЛ, WSFL), наглядно отражающие древовидную структуру процесса.
- Блочные модели (BPML, XLANG), наиболее приближенные к блочной структуре языков программирования.

Проведенный сравнительный анализ двух видов моделей (блочных и графовых) на примере их основных представителей (XPDЛ и BPML) с точки зрения организации в них основных конструкций, определяющих модель потоков работ, показал, что все основополагающие конструкции блочной модели BPML могут быть выражены через их прямые аналоги модели потоков работ XPDЛ. Между тем спецификация XPDЛ обладает рядом очевидных преимуществ:

- Наличие понятия участников процесса (Participants) делает возможным определение человеческих и машинных ресурсов, являющихся исполнителями заданий. При этом выбор исполнителя может определяться

условным выражением по ходу выполнения процесса.

- Спецификация поддерживает расширяемость (определение новых элементов и атрибутов), что позволяет легко адаптировать систему управления потоками работ, основанную на использовании XPDЛ, к различным сферам деятельности, а так же доопределять различные управляющие конструкции.
- Абстрактное понятие внешнего приложения (Application) в совокупности с поддержкой расширяемости стандарта дает свободу выбора в реализации интерфейсов для общения системы управления потоками работ с Web-сервисами, использующимися для выполнения «действий» процесса. Например, в качестве таких интерфейсов может использоваться WSDL-механизм.
- Существует ряд средств визуального проектирования (как графических, так и диалоговых), облегчающих процесс редактирования XPDЛ-описаний.
- Имеется спецификация программных интерфейсов (WAPI) [4], определяющих функциональность системы управления потоками работ, а также модели функционирования распределенных сервисов (ProcessDefinitionInterchangeProcessModel) в рамках объектной модели XPDЛ.
- Ввиду графовой структуры XPDЛ-описаний упрощается процесс импорта описаний процессов на других XML-языках (преобразование их к направленному графу условных переходов между «действиями»), что способствует построению распределенных систем управления потоками работ.

На основании проведенного анализа за базовый стандарт описания модели рабочих процессов по редактированию ресурсов репозитория ИСИР выбрана спецификация XMLProcessDefinitionLanguage (XPDЛ) ввиду наиболее полного соответствия требованиям к функциональности создаваемого решения.

4 Применение языка описания рабочих процессов XPDЛ для реализации службы управления декларируемыми потоками работ по манипулированию ресурсами репозитория Информационного Web-портала РАН

Поскольку Информационный Web-портал РАН является системой, предоставляющей конечному пользователю полную оперативную информацию о произвольной сложности ресурсах РАН, он должен обладать гибкими средствами для поддержания актуальности имеющихся данных: быстрой пакетной загрузки больших объемов информации и автоматизированного управления потоками работ по созданию и редактированию атрибутов существующих в системе ресурсов. XML и RDF загрузка решают задачу о внесении в базу данных больших объемов информации. Редактирование и создание отдельных ресурсов с помощью форм требует наличия системы, управляющей декларируемыми потоками работ пользователей. Даже после пакетной загрузки требуется участие администраторов данных, отвечающих на запросы системы по нарушениям форматов, разрешению неоднозначности, выявлению дубликатов, которое также должно быть осуществляться в рамках ролевых, контролируемых потоков работ.

На базе перечисленных выше требований, была создана служба управления

потоками работ по редактированию ресурсов Информационного Web-портала РАН [11], ядро которого составляет система ИСИР РАН [9], новая версия которой использует платформу Java и ряд opensource решений [8]. Данный сервис использует службу объектно-реляционного отображения, предоставляемую новой версией ИСИР, для хранения объектной модели рабочих процессов в базе данных, а также службу аутентификации и управления пользовательскими правами доступа к объектам процессов. Для декларативного описания новых типов потоков работ используется часть спецификации XPDЛ, обеспечивающая необходимую функциональность. Пользовательские интерфейсы системы управления потоками работ построены на базе Java-технологий ИСИР, автоматизирующих процесс построения Web-форм для редактирования ресурсов. Рассмотрим принципы построения и работу сервиса.

4.1 Архитектура службы управления потоками работ

Общая объектная модель процесса представлена на следующей диаграмме:



[В объектной схеме сервиса, представленной на диаграмме, не отображен еще](#)

один тип объектов – *TransitionHistoryItem* – представляющий собой элемент журнала протоколирования переходов между заданиями. При осуществлении перехода в полях данного объекта сохраняются следующие данные: ссылка на осуществленный переход, комментарий к переходу, ссылка на совершившего его пользователя, дата перехода и т.п. История переходов процесса доступна для просмотра менеджеру процесса через Web-интерфейс.

4.1.1 Рабочий процесс (*WorkflowProcess*)

Объект *WorkflowProcess* представляет собой описание конкретного рабочего процесса. Он имеет ряд атрибутов, задающих текстовые описания процесса, набор ресурсов, для редактирования которых он предназначен, пользователей, имеющих права на создание и управление процессами данного типа, приложение, используемого для выполнения его заданий.

Маршрут процесса представляет собой направленный граф, узлами которого являются объекты *Activity* (задание пользователя), а дугами – объекты *Transition* (переход между заданиями). Процесс может иметь только одно начальное задание и несколько завершающих. Переход по маршруту может осуществляться к одному из следующих в соответствии с декларацией процесса заданий или к любому из предыдущих (включая начальное). В каждый момент времени может существовать не более одного «активного» пользовательского задания для данного процесса. Все остальные задания (узлы маршрута) должны иметь статус «неактивно» или «завершено».

Процесс может находиться в одном из трех *состояний*: «активен», «неактивен» или «завершен». Если процесс находится в «активном» состоянии, его текущее «активное» задание доступно исполнителям. В «неактивном» и «завершенном» состоянии никакие задания процесса не видны его исполнителям. Помимо этого, статус процесса влияет на автоматическое управление датами создания и завершения процесса. Управление статусами процесса выполняется системой или администратором через соответствующие Web-интерфейсы. «Активное» состояние процесса разбивается на два дополнительных состояния: «активен и просрочен» и «активен и не просрочен». Просроченным процессом считается любой процесс, активное задание которого не укладывается в отведенные для него временные рамки.

В роли редактируемых с помощью данного процесса ресурсов могут использоваться произвольные хранимые объекты «ядра» портала. Любой тип процесса может применяться как к произвольным объектам системы, так и к выделенному набору типов ресурсов, для которых должно существовать более специфичное описание потоков работ.

4.1.2 Задание процесса (*Activity*)

Задание процесса *Activity* является точкой маршрута процесса, в которой осуществляются работы по изменению содержания ресурса. На каждое задание накладываются строгие временные рамки, представленные тремя датами: дата постановки (активации) задания, дата крайнего срока исполнения задания, дата фактического завершения выполнения задания.

Управление всеми перечисленными датами может выполняться как в автоматическом режиме (системой) при переходах между заданиями, так и администратором процесса.

В XML-декларации задания имеется ссылка на приложение, которое предоставляет пользователю визуальные интерфейсы для выполнения этого задания. В соответствии со спецификацией XPDЛ информация о типе конкретного приложения не включается в описание процесса. Задание должно знать только о том, куда направить пользователя для редактирования ресурса; весь процесс редактирования контролируется самим приложением. В существующем применении данной службы для управления потоками работ по созданию новых ресурсов Информационного Web-портала РАН для модификации атрибутов ресурсов в узлах маршрута используется служба «ядра» для управления формами редактирования ресурсов и ее понятие *wizard-a*. Wizard представляет собой группу форм, служащую для редактирования одного или нескольких ресурсов, отличительной чертой которой является назначение прав на отдельные формы (это отражается и в навигационном меню *wizard-a*). Это позволяет организовать более гибкое управление ходом выполнения потоков работ.

Каждому заданию в декларации процесса назначается один или более исполнителей. Исполнители имеют права на чтение информации о задании, запуск приложения для выполнения задания и переход к следующему заданию в соответствии с правами доступа на переходы. В роли исполнителя задания может выступать отдельно взятый пользователь или группа пользователей (это декларируется в XML-описании процесса). При определении пользователей и групп, а также при назначении прав на объекты процесса используются механизмы и сервисы, предоставляемые «ядром».

Как и в случае всего процесса, каждое из его заданий может иметь следующий набор состояний: «активно» («просрочено» или «не просрочено»), «неактивно» или «завершено». Управление статусами заданий выполняется либо системой, либо администратором процесса.

4.1.3 Переход между заданиями (Transition)

Направленные дуги переходов между заданиями маршрута процесса представлены объектами *Transition*. Для перехода существует условное понятие направления («прямое» или «обратное»), которое влияет на управление характерными датами заданий, между которыми осуществляется переход. При переходе от одного задания к другому системой выполняется следующая последовательность действий:

- Если происходит переход в прямом направлении, то текущее активное задание завершается (проставляется статус «завершено», дата завершения, выполнивший его пользователь), а следующее (указанное в декларации) задание активируется (проставляется статус «активно», и дата постановки инициализируется текущей датой).
- Если происходит переход в обратном направлении, то текущее активное задание деактивируется (проставляется статус «неактивно»), а одно из предыдущих заданий, к которому осуществляется переход (в данный

момент завершено) активируется (проставляется статус «активно», но дата постановки не сбрасывается). Таким образом, если происходит возврат к одному из завершенных заданий, то для этого задания все временные рамки остаются такими же, как и до завершения (в том числе и крайний срок исполнения). Данная политика означает то, что задание считается полностью выполненным только тогда, когда на него нет возвратов ввиду его неверного исполнения.

- Если задание, к которому должен осуществляться переход не указано, то текущее задание считается последним в маршруте, и происходит завершение всего процесса – изменяется статус процесса и проставляется дата завершения.
- Информация о совершенном переходе заносится в журнал переходов процесса.

На все переходы между заданиями (так же как и на сами задания) назначаются права доступа. При отсутствии явно указанных исполнителей конкретного перехода, он становится доступен всем участникам процесса, имеющим доступ к заданию, с которого осуществляется этот переход.

4.2 Формат для описания рабочих процессов

Структура XML-документа для декларации процессов построена в соответствии со спецификацией XPDЛ, но является сокращенной по сравнению со своим прототипом ввиду применения в более конкретных областях, а значит и с более ограниченной функциональностью. Эта необходимая часть структуры XPDЛ-описания выбиралась с таким расчетом, чтобы поддержать всю функциональность, требуемую от системы управления потоками работ на данном этапе развития Java-проекта, а в дальнейшем (возможно, с некоторыми дополнениями) и для применения в других системах использующих понятие рабочих процессов – таких как, например, системы электронного документооборота. Ввиду применения в более конкретной области построения Web-порталов к XML-декларации были добавлены некоторые дополнительные элементы.

Реализованная модель рабочих процессов предполагает наличие трех predetermined пользовательских ролей:

- **MANAGER** – менеджер процесса. Может менять статусы и удалять экземпляры процессов, менять статусы заданий, изменять их комментарий и продолжительность, добавлять и исключать пользователей из списка их исполнителей – то есть выполнять администраторские функции. Может являться исполнителем заданий и осуществлять переходы.
- **CREATOR** – может создавать новые экземпляры процессов, но не имеет возможностей менеджера по управлению процессами. Может являться исполнителем заданий и осуществлять переходы.
- **PERFORMER** – является исполнителем доступных ему заданий и выполняет назначенные для него переходы.

Первые две роли жестко декларируются в XML-описании процесса и не могут быть

назначены (или сняты) у пользователя в ходе его выполнения. Последняя роль может динамически назначаться (сниматься) менеджером процесса с помощью Web-интерфейса.

4.3 Применение службы управления потоками работ в системах электронного документооборота

В данный момент широко распространены системы электронного документооборота, предназначенные для применения в различных областях: банковская сфера, документооборот промышленного предприятия, учебных заведений и т.д. Поскольку «ядром» подобных систем является система управления потоками работ, снабженная некоторыми специфичными для конкретной предметной области понятиями и функциональностью, то возникает естественный вопрос о степени применимости имеющегося решения для использования в управлении потоками работ с документами.

С этой целью был проведен анализ соответствующих требований к системам, подготовлено сопоставление основных понятий, систем электронного документооборота и имеющейся реализации системы управления потоками работ. Приведенный анализ показывал, что применение службы управления потоками работ в системах электронного документооборота требует поддержки некоторых дополнительных возможностей, специфичных для этой области, например, фиксация операционных состояний документа. Кроме того, ряд проблем обусловлен неполнотой текущей реализации, в которой еще не поддерживаются некоторые ключевые моменты, например, полноценное ведение версий, поддержка транзакций.

Практичность имеющегося на текущий момент решения, выявленные дополнительные потребности свидетельствуют о необходимости дальнейшего развития системы управления потоками работ, реализации на ее основе простой системы электронного документооборота, которая будет востребована в WCMS.

4.4 Оценка текущей реализации службы управления потоками работ

Проектирование и разработка системы управления потоками работ по манипулированию ресурсами репозитория ИСИР разбита на несколько этапов. К данному моменту (частично или полностью) поддерживается следующая функциональность:

1) **Выбор необходимого подмножества конструкции XPDL.** Поддержаны основные структуры XPDL, необходимые для реализации рабочих процессов по созданию и редактированию ресурсов репозитория ИСИР. Пока нет поддержки составных заданий и подпроцессов, а так же ряда других второстепенных элементов. Добавлены некоторые дополнительные элементы и атрибуты для задания временных ограничений, локализованных наименований и т.д. Не реализована поддержка пакетов описаний с возможностью их экспорта и импорта.

2) Создание основных компонентов объектной модели процесса в соответствии со спецификацией XPDL. Реализованы Java-классы, соответствующие объектам Transition, WorkflowProcess и Activity метамодели XPDL с необходимым набором атрибутов и некоторыми методами, расширяющими их функциональность. Данная модель состоит из хранимых в базе данных объектов, что обеспечивается объектно-реляционным отображением, базирующемся на службах «ядра». В качестве участников (Participant) процесса сейчас выступают пользователи «ядра» системы. Внешние приложения пока представлены специальным Java-интерфейсом и обязаны являться Java-классами. На данный момент не поддерживаются некоторые второстепенные компоненты объектной модели, такие как хранилище ресурсов (ResourcesRepository), системные данные и переменные окружения (SystemandEnvironmentData) и т.д. В текущей версии переходы между заданиями являются безусловными.

3) Разработка и реализация компонента системы, создающего хранимые модели процессов по их XPDL-описанию. Реализован разбор XPDL-документов с помощью DOM-модели и использования XPath и создание хранимой в базе данных объектной модели процесса с инициализированными начальными значениями атрибутами. При создании объектов происходит назначение на них прав доступа в соответствии с XPDL-декларации и с использованием служб «ядра», управляющих проверкой и назначением прав доступа на хранимые объекты.

4) Разработка подсистемы создания и управления ходом выполняемого процесса.

Реализован набор функций по созданию управлению процессами по ходу их выполнения, а так же по получению информации о заданиях и процессах. К основным из них можно отнести:

- Создание новых экземпляров процессов.
- Получение списков процессов, заданий и переходов в соответствие с правами текущего системного пользователя.
- Осуществление перемещений по маршруту процесса.
- Изменение состояний процессов и заданий.
- Удаление и завершение процессов и заданий.
- Обновление ряда управляющих атрибутов компонентов процесса.
- Управление исполнителями заданий и переходов.
- Запуск внешних приложений, отвечающих за выполнение заданий (на данный момент такие приложения являются Java-классами).

5) Реализация подсистемы контроля и протоколирования. Контроль выполнения заданий осуществляется посредством отведения на них временных рамок. Предусмотрены средства для уведомления исполнителей и системных администраторов о «просроченных» заданиях. Назначение временных рамок на процесс в целом пока отсутствует. Элемент протокола управляющих операций (переходов между заданиями) сейчас представлен специальным хранимым объектом TransitionHistoryItem, который сохраняет в себе основную информацию о совершенном переходе. Протоколирование других типов операций пока не поддержано. Информация о сроках фактического выполнения задания,

выполнившем его пользователе сейчас хранится непосредственно в атрибутах самого объекта Activity. Поддержка других физических носителей для ведения системных протоколов, отличных от реляционной базы данных, пока не предусмотрена.

6) Интеграция с базовыми службами управления доступом к хранимым объектам и аутентификацией пользователей. На данный момент все компоненты хранимой модели рабочего процесса защищены от несанкционированного доступа посредством базовых служб ИСИР [9]. При этом в системе выделено несколько предопределенных пользовательских ролей, назначаемых участникам процесса в XPDL-описании. Для аутентификации пользователей при входе в систему используется сервис аутентификации ядра ИСИР.

7) Разработка Web-интерфейсов для создания и управления потоками работ, а также для администрирования системы. Реализованы управляющие Web-интерфейсы службы управления потоками работ на базе сервиса, автоматизирующего процесс построения форм редактирования и страниц просмотра атрибутов хранимых объектов с помощью технологии JSP-шаблонов. Помимо этого на базе этой же службы реализованы Web-приложения (wizard-ы) для редактирования ресурсов репозитория ИСИР – внешние приложения, использующиеся службой управления потоками работ для выполнения заданий процессов. Работа сервиса продемонстрирована на примере описания потоков работ двух различных типов.

4.5 Следующие шаги по развитию системы

К ближайшим шагам по дальнейшей разработке и модернизации компонентов системы можно отнести следующее:

- Дополнение выбранного подмножества элементов XPDL конструкциями, поддерживающими рекурсивные процессы и задания (например, группы заданий BlockActivity и ActivitySet, а так же подпроцессы SubFlow).
Расширение набора управляющих атрибутов созданной объектной модели.
- Реализация модели пакетов XPDL с возможностью экспорта и импорта элементов описания.
- Добавление остальных компонентов объектной модели XPDL с поддержкой их хранения в реляционной базе данных.
- Поддержка исключительных ситуаций и их обработчиков с возможностью оповещения как исполнителей, так и администратора системы. Реализация как системных, так и пользовательских обработчиков исключений.
- Реализация некоторого унифицированного интерфейса для общения системы с внешними Web-приложениями произвольного типа (возможно, базирующегося на описании интерфейсов Web-служб с помощью WSDL).
- Поддержка условных переходов между заданиями и условных циклических конструкций языка описания рабочих процессов XPDL.
- Поддержка множественных переходов с одного задания (параллельный запуск нескольких заданий с возможностью их синхронизации).
- Реализация подсистемы управляющей версиями хранимых объектов,

поддерживающей как ведение версий отдельных атрибутов объекта, так и объекта целиком с учетом связей с другими ресурсами.

- Реализация подсистемы управления транзакциями заданий на базе службы ведения версий хранимых объектов.
- Усовершенствование подсистемы протоколирования операционных состояний объектов рабочего процесса. Добавление возможности ведения протоколов на различных физических носителях. Интеграция со службой аудита хранимых объектов «ядра».
- Модернизация Web-интерфейсов системы управления потоками работ.
- Выбор и применение одного из существующих средств визуального проектирования для создания XPD-описаний рабочих процессов.

К задачам второстепенной важности можно отнести поддержку возможности экспорта XPD-описаний с предварительным преобразованием к другим существующим стандартам декларации потоков работ и реализацию распределенной системы управления рабочими процессами.

Заключение

В наше время происходит стремительный рост числа информационных Web-порталов и цифровых библиотек, в которых все большее и большее применение находят автоматизированные системы управления потоками работ. С их помощью становится возможным увеличение качества и объемов выполняемых работ по манипулированию информационным наполнением и сервисами Web-порталов и цифровых библиотек.

Поддержка и следование службой управления рабочими процессами по манипулированию ресурсами репозитория ИСИР ряду соответствующих стандартов делает возможным ее применение в системах управления контентом Web-порталов, системах электронного документооборота и других сервисах, автоматизирующих управление потоками работ, распределенных между разнообразными участниками производственных процессов. Широчайшая область применения служб подобного типа дает не менее широкие предпосылки к дальнейшему развитию и модернизации данного проекта.

Литература

[1] BPML working draft March 25, 2002.

<http://www.bpmi.org>

<http://xml.coverpages.org/bpml.html>

[2] Robert Shapiro. A Comparison of XPD, BPML and BPEL4WS.

<http://xml.coverpages.org/Shapiro-XPD.pdf>

[3] RDF Vocabulary Description Language 1.0, RDF Schema. W3C Working Draft 23 January 2003.

<http://www.w3.org/TR/rdf-schema>

[4] Workflow Management Coalition standards.

<http://www.wfmc.org/standards/standards.htm>

- [5] Workflow Process Definition Interface – XML Process Definition Language.
WFMC-TC-1025 -http://www.wfmc.org/standards/docs/TC-1025_10_xpdl_102502.pdf
- [6] Web Services Flow Language.
<http://www-4.ibm.com/software/solutions/webservices/pdf/WSFL.pdf>,
<http://xml.coverpages.org/wsfl.html>
- [7] Web Services for Business Process Design.
http://www.gotdotnet.com/team/xml_wsspecs/clang-c,
<http://xml.coverpages.org/clang.html>
- [8] Бездушный А.Н., Жижченко А.Б., Кулагин М.В., Серебряков В.А.
Интегрированная система информационных ресурсов РАН и технология
разработки цифровых библиотек. Программирование V 26, N 4, 2000, pp. 177-185
- [9] Бездушный А.А., Сысоев Т.М., Нестеренко А.К., Бездушный А.Н., Серебряков
В.А., RDFS как основа среды разработки цифровых библиотек и Web-порталов //
Электронные библиотеки, 2003, Том 6, Выпуск 3.
<http://www.elbib.ru/index.phtml?page=elbib/rus/journal/2003/part3/BBNSS>
- [10] Бездушный А.А., Сысоев Т.М., Нестеренко А.К., Бездушный А.Н., Серебряков
В.А., Архитектура и технологии RDFS-среды разработки цифровых библиотек и
Web-порталов // Электронные библиотеки, 2003, Том 6, Выпуск 4.
<http://www.elbib.ru/index.phtml?page=elbib/rus/journal/2003/part4/BNSBS>
- [11] Концепция создания Единой информационной системы РАН (вторая
редакция). http://uis.isir.ras.ru/win/htm/scientific_activity.html?p=5p7p2
-

Об авторах

Бездушный Алексей Анатольевич - студент Московского Физико-Технического института. Сфера деятельности: программирование, Java, распределенные системы, RDFS, Semantic Web, онтологии, Web-порталы, базы данных, цифровые библиотеки.

Тел. (095)1355471

E-mail: alix@7ka.mipt.ru

Нестеренко Алексей Константинович - аспирант Вычислительного Центра имени А. А. Дородницына РАН. Сфера деятельности: программирование, Java, управление потоками работ, базы данных, Web-порталы, цифровые библиотеки.

Тел. (095)1355471

E-mail: alexn@ccas.ru

Сысоев Тимофей Михайлович - аспирант Вычислительного Центра имени А. А.

Дородницына РАН. Сфера деятельности: программирование, Java, распределенные системы, атрибутно-полнотекстовый и распределенный поиск, Web-порталы, базы данных, цифровые библиотеки.

Тел. (095)1355471

E-mail: tim@ccas.ru

Бездушный Анатолий Николаевич - кандидат физико-математических наук, с.н.с. Вычислительного Центра имени А. А. Дородницына РАН. Сфера деятельности: системное программирование, параллельные вычисления, сети, базы данных, распределенные системы, информационно-поисковые технологии, цифровые библиотеки, Web-порталы.

Тел. (095)1355471(*4216)

E-mail: bezdushn@ccas.ru

Серебряков Владимир Алексеевич - доктор физико-математических наук, профессор, зав.отделом Вычислительного Центра имени А. А. Дородницына РАН. Сфера деятельности: системное программирование, параллельные вычисления, сети, базы данных, распределенные системы, информационно-поисковые технологии, цифровые библиотеки, Web-порталы.

Тел. (095)1355471(*4220)

E-mail: serebr@ccas.ru
