

ISTC 1993 P

**Годовой технический отчет по проекту  
ISTC 1993 P**

**Математические основы обнаружения знаний в данных и автономные  
интеллектуальные архитектуры**

**(С 1 декабря 2001 по 1 декабря 2002, 12 месяца)**

*Городецкий Владимир Иванович*  
**(Менеджер проекта)**

**Санкт-Петербургский институт информатики и автоматизации  
Российской академии наук**

**Партнер: European Office of Aerospace Research and Development**

**Декабрь 2002**

Менеджер проекта  
Главный научный сотрудник  
Санкт-Петербургского института  
информатики и автоматизации  
Доктор технических наук  
профессор

**В.И.Городецкий**

## Задача 1

### 1.1 Название проекта. Номер годового отчета

Проект 1993Р: Математические основы обнаружения знаний в данных и автономные интеллектуальные архитектуры.

Задача 1: Разработка методов извлечения знаний из баз данных и разработка программного прототипа многоагентной системы обучения объединению данных

Отчет №2

### 1.2. Головной институт

Санкт-Петербургский институт информатики и автоматизации Российской академии наук

### 1.3. Институты-участники

Нет

### 1.4. Руководитель проекта, номер телефона, факса, адрес электронной почты

Городецкий Владимир Иванович, (812)-323-3570, (812)-328-0685, gor@mail.iias.spb.su

### 1.5. Дата начала осуществления, продолжительность проекта

1 декабря 2000, 36 месяцев

### 1.6. Краткое описание плана работ: цель, предполагаемые результаты, научно-технический подход

*Краткий план работ*

А-1. Концептуальный анализ технологий объединения данных на основе знаний и их компонент. Разработка концептуальной модели системы обучения объединению данных.	1-4 кварталы
<i>Промежуточный отчет #1, представляющий частичные результаты исследований по задаче А-1.</i>	2 квартал
<i>Промежуточный отчет #2, представляющий результаты исследований по задаче А-1.</i>	4 квартал
<i>Представление статьи в международный журнал.</i>	5 квартал
А-2. Разработка архитектуры и объектно-ориентированного проекта многоагентной системы обучения объединению данных.	5-8 кварталы
<i>Промежуточный отчет #3, представляющий результаты исследований по задаче А-2.</i>	8 квартал
А-3. Разработка программных прототипов компонент системы обучения объединению данных.	9-11 кварталы
<i>Итоговый отчет, содержащий результаты по данной задаче проекта в целом, а также заключение.</i>	11 квартал
<i>Демонстрация разработанных программных компонент системы обучения объединению данных. (По требованию AFRL/IF)</i>	12 квартал

Примечание: Строки таблицы, залитые серым цветом, отвечают исследованиям, запланированным на второй год работы.

*Цель проекта* состоит в разработке математических методов, формальной модели, архитектуры и программного прототипа многоуровневой обучающей системы, построенной на основе технологии многоагентных систем для поддержки разработки распределенной базы знаний системы объединения данных из различных источников.

### *Ожидаемые результаты*

1. Онтология подзадач, которые необходимо решать в рамках задачи объединения данных.
2. Множество математически корректных методов обнаружения знаний в базах данных, которые обеспечивают решение задачи разработки баз знаний систем объединения данных.
3. Математические методы распределенного обучения на основе технологии интеллектуальных агентов, объединенных в рамках многоагентных систем
4. Математические методы объединения знаний, полученных из локальных источников.
5. Множество программно реализованных интеллектуальных агентов, реализующих методы извлечения знаний из распределенных баз данных-прецедентов.
6. Пользовательский интерфейс, обеспечивающий возможность интерактивной разработки системы распределенного обучения применительно к конкретному приложению.
7. Результаты компьютерного моделирования разработанной модели обучения, реализованной в виде программного прототипа.

### *Научно-технический подход*

При исследованиях математического характера используются известные подходы к извлечению знаний из данных на базе методов математической статистики и дискретной математики. Кроме того, предполагается использовать оригинальные методы. В частности, будет использован визуальный подход к извлечению правил из числовых данных, методы извлечения таких правил из разнотипных данных типа булевых, категориальных, упорядоченных и целочисленных. При разработке многоуровневой архитектуры распределенного обучения предполагается использовать методологию мета-классификации, разработанную школой С.Столфо (США). Будут использованы также методы теории вероятностей, дискретной математики, математической логики, прикладные методы искусственного интеллекта, в частности, методы инженерии знаний и машинного обучения, теории протоколов и переговоров и др.

При разработке архитектуры программного обеспечения используются современные достижения в области многоагентных систем. При разработке программного продукта используется современный подход на основе объектно-ориентированного проектирования с применением одного из стандартных инструментальных средств.

Программная реализация прототипа многоагентной системы обучения, а также компоненты системы объединения данных разрабатываются с использованием современных средств поддержки программных разработок, в частности, с использованием Visual C++ 6.0, Java 2, SQL Server и XML.

В целом, при исследованиях в рамках данной задачи используются методологические подходы, принятые в области искусственного интеллекта, многоагентных систем, в области систем, основанных на знаниях, машинного обучения, объектно-ориентированного проектирования, а также методы "классической" математики.

## **1.7. Ход выполнения технических работ за первый год (для годовых отчетов за второй год)**

## **1.8. Ход выполнения технических работ за рассматриваемый год**

*Ход выполнения работ* полностью соответствует плану работ как по содержанию, так и по срокам завершения предусмотренных этапов работ.

### *Основные достижения за прошедший год*

В соответствии с Планом работ на второй год исследований, основные исследования проводились по задаче А-2: "Разработка архитектуры и объектно-ориентированного проекта многоагентной системы обучения объединению данных." Эта задача включает в себя следующие подзадачи:

1. Выявление типовых подзадач задачи обучения объединению данных и их распределение между типовыми классами агентов системы обучения.
2. Разработка архитектур типовых агентов системы обучения, реализующих выбранные и разработанные алгоритмы извлечения знаний из данных.
3. Разработка модели и архитектуры взаимодействия агентов в процессе распределенного обучения. Разработка методов и архитектуры объединения знаний, полученных из данных, и экспертных знаний.

4. Разработка компонент модели коммуникаций агентов, включая язык и протокол обмена сообщениями.
5. Разработка объектно-ориентированного проекта компонент многоагентной системы обучения объединению данных и интерфейсов их взаимодействия

Результаты исследований по названным задачам были полностью представлены партнеру в Промежуточном отчете №3 к 1 декабря 2002 г. Краткое резюме основных результатов приводится далее.

*1. Выявление типовых подзадач задачи обучения объединению данных и их распределение между типовыми классами агентов системы обучения.*

На основе концептуального анализа задачи обучения в системах слияния данных был составлен полный список задач, которые необходимо решать в процессе обучения, а также их распределение между типовыми классами агентов, формирующими архитектуру инструментария для разработки систем слияния данных. Архитектура инструментального средства включает в себя компоненты двух типов:

- (1) Компоненты, которые расположены на тех же хостах, на которых хранятся данные источников. Эти компоненты ответственны за поддержку технологии разработки локальных компонент системы слияния данных;
- (2) Компоненты, которые ответственны за разработку подсистемы управления процессом обучения. Эта компонента решает задачи мета-уровня.

Соответствующим образом разделяются и агенты инструментального средства, т.е. часть из них отвечает за работу с локальными источниками данных, а другая часть решает задачи мета-уровня. Архитектуры каждой из названных компонент и формирующие их типы агентов представлены на рис 1 и 2.

Укажем основные типы задач инструментальной системы слияния данных и их распределение по агентам, не вдаваясь в детализацию содержания самих задач. Последний аспект представлен в Промежуточном отчете №3. Названия задач соответствуют протоколу, представленному на рис.4.

*Агент-Мастер обучения*

Поддержка разработки распределенной онтологии приложения.

Поддержка разработки мета-модели принятия решений системой слияния данных в целом.

Управление распределенным обучением.

*Агент мета-обучения*

Управление распределенным обучением.

Поддержка разработки мета-модели принятия решений системой слияния данных в целом.

Рассылка разработанных структур принятия решений агентам локального уровня.

*Агент принятия решений (классификации) на мета-уровне*

Управление распределенным обучением.

Объединение решений классификаторов базового уровня (уровня источников данных).

*Агент управления слиянием данных*

Участие в процессе разработки мета-модели принятия решений системой слияния данных в целом.

Участие в управлении распределенным обучением.

Управление объединением решений классификаторов уровня источников данных.

*Агент обучения базовых классификаторов*

Участие в разработке мета-модели принятия решений системы слияния данных в целом.

Управление обучением базовых классификаторов.

*Агент принятия решений (классификации) локального источника данных*

Управление принятием решений базовыми классификаторами

Реализация функций принятия решений отдельными базовыми классификаторами.

*Агент управления данными локального источника*

Участие в процессе принятия решений.

Участие в разработке распределенной онтологии.

Участие в разработке мета-модели принятия решений системы слияния данных в целом.

Участие в процессе принятия решений.

Мониторинг источников данных с целью анализа наличия новых данных.

## *2. Разработка архитектур типовых агентов системы обучения, реализующих выбранные и разработанные алгоритмы извлечения знаний из данных.*

Опыт показывает, что при разработке любой новой прикладной многоагентной системы необходимо решать задачи двух различных типов. Задачи одного из них являются общими для значительного числа приложений и не связаны непосредственно с предметной областью, к которой относится разрабатываемое приложение, в то время как задачи другого типа являются предметно-ориентированными. В соответствии с современными взглядами, для решения задач первого типа целесообразно использовать инструментальные средства, которые должны предоставлять готовые или почти готовые решения для инвариантных проблем, освобождая тем самым разработчиков конкретных приложений от значительного объема рутинной работы. При наличии адекватных задач инструментальных средств разработчики имеют возможность концентрировать свои усилия на решении задач предметного характера.

Инструментальные системы должны предлагать разработчикам некоторые инвариантные компоненты архитектуры агентов различного назначения. Если такие компоненты объединены в рамках общей программы, и эта программа играет роль общей части множества различных прикладных агентов, то ее принято называть термином "*типовой агент*" ("*generic agent*"). Эту программу, содержащую инвариантные программные компоненты "*будущих*" прикладных агентов, следует рассматривать как уже готовую программу агента, подлежащего специализации в соответствии с его назначением и функциональностью в МАС. Такая программа должна объединять библиотеки методов работы с базами знаний агентов, а также некоторые заранее заданные, однако "пустые" структуры для представления знаний о предметной области, согласованные между собой. При наличии соответствующих инструментальных средств использование таких методов описания фрагментов знаний и наполнение "пустых" структур баз знаний становится основным содержанием разработки прикладных агентов.

Следуя описанной выше идее, в принятой технологии любой прикладной агент состоит из двух частей: *Инвариантной платформы для разработки агента* (это другое название для "*generic agent*", "*типовой агент*", используемое наравне с ним) и *Прикладной компоненты*. Первая из них реализует совокупность функций, инвариантных по отношению к предметной области и является общей (одинаковой) для всех агентов прикладной системы. Вторая компонента содержит спецификации конкретных функциональностей агента, сценариев его поведения, его предметных знаний и данных, конкретных механизмов работы с ними, спецификации протоколов и т.п.

Рис.3 представляет концептуальную модель прикладного агента, которая формируется из названных двух компонент, а также поясняет схему их взаимодействия. Детальное описание разработанного *типового агента*, а также его использование в технологии разработки прикладных агентов дано в Промежуточном отчете №3, представленном Партнеру в ноябре 2002 г.

## *3. Разработка модели и архитектуры взаимодействия агентов в процессе распределенного обучения. Разработка методов и архитектуры объединения знаний, полученных из данных и экспертных знаний*

Ключевые проблемы разработки и программной реализации инструментального средства для поддержки технологии систем слияния данных обусловлены особенностями данных, которые используются для обучения и принятия решений, и прежде всего—их гетерогенностью и распределенностью. Как правило, эти данные хранятся в распределенных базах данных и характеризуются разнообразием физической природы, шкал измерения и структур представления, неполнотой и неопределенностью и т.д. Эти свойства данных оказывают основное влияние как на структуру выработки решений, так и на архитектуру системы обучения, а также влекут необходимость решения ряда новых проблем.

Среди последних проблем следует выделить две основные, решение которых выполняется в распределенном варианте на основе кооперации различных компонент (агентов) инструментария. Первая проблема—это формирование мета-модели (структуры) объединения решений, принимаемых на основе данных локальных источников, и управление распределенным обучением системы слияния данных. Вторая проблема обусловлена необходимостью формирования мета-модели распределенных источников данных, которая необходима для того, чтобы обеспечить "*единый взгляд*" на гетерогенные данные, вовлекаемые в процесс обучения и принятия решений, устранить противоречия и неоднозначности в представлении данных в различных источниках. Эти проблемы были в центре внимания в течение второго года исследований. Соответствующие результаты детально представлены в Промежуточном отчете №3. Они включают в себя

концептуальный анализ названных проблем, разработанные подходы к их решению, а также модели координации поведения различных агентов инструментария, вовлеченных в формирование мета-модели обучения и принятия решений, в также мета-модели источников данных. Высокоуровневый протокол взаимодействия агентов инструментария поддержки технологии разработки систем слияния данных представлен на рис.4 в форме стандартной IDEF0 диаграммы. Отметим, что центральными процедурами в процессе взаимодействия агентов здесь являются процедуры формирования распределенной онтологии и распределенное обучение. Последняя включает в себя обучение базовых классификаторов источников и объединение их решений на мета-уровне. Заметим, что все компоненты протокола, представленного на рис.4, проработаны до степени детализации, которые отвечают действиям, реализуемым уже отдельными неделимыми алгоритмами, выполняемыми конкретными агентами.

#### *4. Разработка компонент модели коммуникаций агентов, включая язык и протокол обмена сообщениями.*

Структурная схема коммуникации агентов, реализованная в рамках данного проекта для многоагентных систем слияния данных, представлена на рис.5. В ней обмен сообщениями между любой парой агентов поддерживается тремя промежуточными компонентами:

*порталом* компьютера, на котором находится агент–источник сообщения,  
*мета-агентом* коммуникаций (он один на всю многоагентную систему) и  
*порталом* компьютера, на котором расположен агент–адресат.

Протоколы коммуникаций, используемые в разрабатываемой схеме обмена сообщениями, делятся на три группы:

1. Протоколы, которые поддерживают стандартный обмен сообщениями по общепринятой (для многоагентных систем) трехуровневой схеме: "стандартный транспортный протокол"–синтаксис сообщения"–"содержание сообщения".

2. Протоколы, которые предназначены для реализации множества семантически взаимосвязанных диалогов агентов, возникающих при решении агентами различных подзадач, требующих использования сложных диалогов. Данные протоколы играют роль мета–протоколов по отношению к протоколам первой группы.

3. Технологические протоколы, которые предназначены для обеспечения взаимодействия агентов при распределенном решении некоторых содержательных задач. Протоколы этого типа играют роль мета–протоколов по отношению к протоколам первого и второго типов

*В протоколах первой группы* транспортный уровень реализуется в проекте на основе стандартной технологии JAVA RMI (*Remote Method Invocation*), которая базируется на использовании стандартного протокола TCP/IP; синтаксис сообщений представляется на языке KQML, а семантика сообщения пока представляется с помощью языка XML, однако в дальнейшем планируется переход на один из языков RDF или DAML после того, как будут проведены соответствующие работы, связанные с поддержкой разработки онтологии в терминах одного из них. В целом, протоколы данного типа предназначены для кодирования, транспорта, приема и понимания каждого конкретного сообщения.

*Протоколы второй группы* пока представлены в терминах множества схем обработки заданного конечного множества сообщений. В принципе, это позволяет реализовать все необходимые схемы взаимосвязанных диалогов между агентами. Заметим, что каждый "диалог" между парой агентов может включать в себя активацию целой серии диалогов агента-получателя с другими агентами, что может быть необходимо для выполнения задания, содержащегося в сообщении. В свою очередь, у агентов-получателей вторичных сообщений может возникнуть потребность обратиться к другим агентам и т.д. Протоколы второй группы имеют целью поддержку таких многоступенчатых взаимосвязанных диалогов. В настоящее время ведется разработка формальной модели спецификации диалогов и соответствующего инструментария для их описания (хотя это и не предусматривается планом работ по проекту), в которой все множество диалогов агентов (в рамках конкретного приложения) будет представлено с помощью единой вычислительной модели. Это позволит значительно повысить гибкость многоагентной системы при введении в нее изменений, в частности, здесь идет речь о внесении изменений в схемы диалогов агентов.

*Протоколы третьей группы* не являются общими для всех многоагентных систем, как это имеет место по отношению к протоколам первых двух групп. Они необходимы именно в системах объединения данных. В частности, необходимо использовать целую серию специализированных

протоколов, поддерживающих процессы распределенного формирования компонент онтологии распределенной системы объединения решений. Эти протоколы перечислены на рис.4, где они, в свою очередь, рассматриваются как компоненты высокоуровневого протокола. Заметим, что протоколы третьей группы составляют наиболее ответственную часть в обеспечении распределенной работы системы объединения данных. Эти протоколы к настоящему времени полностью разработаны и реализуются программно.

#### *5. Разработка объектно-ориентированного проекта компонент многоагентной системы обучения объединению данных и интерфейсов их взаимодействия*

Объектно-ориентированный проект компонент инструментального средства поддержки технологии разработки систем слияния данных к настоящему времени разработан. Он включает в себя архитектуру (см. рис.1, 2), распределение задач между различными агентами (эти задачи и их распределение были перечислены ранее в п.1, представляющем результаты решения задачи А-2.1), протоколы, перечисленные в предыдущем пункте (протокол верхнего уровня представлен на рис.4), а также описание протоколов в нотациях языка UML в терминах *Uses cases*–диаграмм, *Collaboration*–диаграмм, *State-chart*–диаграмм и *Component* –диаграмм. В целом они обеспечивают полную спецификацию разрабатываемого инструментария, необходимую для написания программного кода. Все названные типы диаграмм, составляющие объектно-ориентированный проект инструментария, были представлены Партнеру в Промежуточном отчете №3.

### **1.9. Существующее положение дел с выполнением технических работ**

Ход выполнения работ полностью соответствует предусмотренному плану и в коррекции не нуждается.

### **1.10. Сотрудничество с зарубежными партнерами**

В соответствии с Рабочим планом, 1 декабря 2002 года Партнеру был представлен Промежуточный отчет №3. Он содержит описание всех результатов, предусмотренных планом исследований. Руководитель работ по Задаче 1, В.И.Городецкий, принимал участие в специальной сессии, организованной Air Force Office of Scientific Research, названной "*Исследовательская инициатива Air Force в области слияния данных*", которая проводилась Партнером в рамках Международной конференции "Fusion-2002" (3-6 июля 2002 года, Annapolis, США). На этой сессии им был представлен доклад по результатам исследований в рамках задачи 1 проекта.

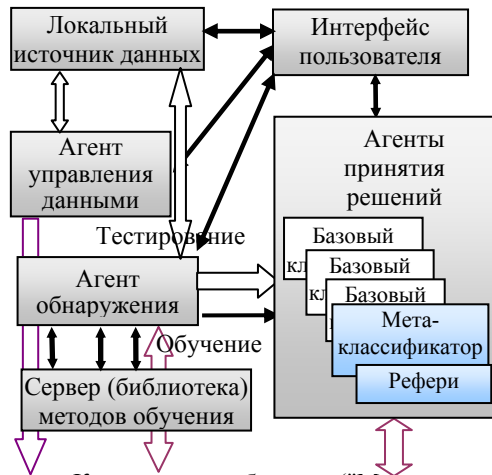
### **1.11. Выявленные проблемы и предложения относительно их устранения**

Нет

### **1.12. Перспективы дальнейшего развития разработанной технологии/научного исследования**

Перспективы дальнейшего сотрудничества будут обсуждаться на встрече с представителями Партнера и Министерства обороны США ориентировочно в апреле 2003. Предложения по дальнейшему сотрудничеству были представлены Партнеру в сентябре 2002.

## Приложение 1.1. Наглядные материалы, прилагаемые к основному тексту



К агенту мета-обучения ("Мастеру")  
Рис.1. Архитектура компоненты системы обучения, работающей на локальном источнике

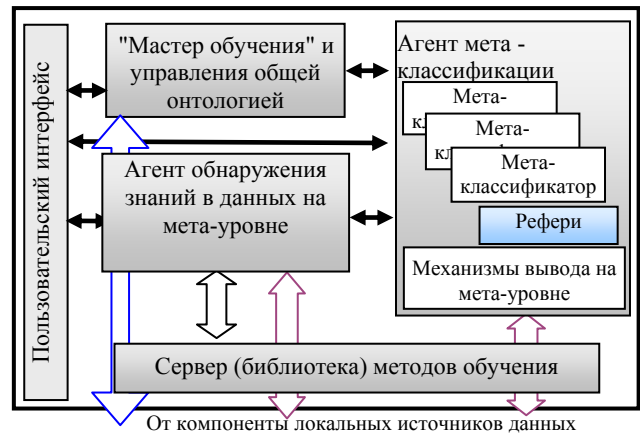


Рис.2. Архитектура многоагентной системы обучения объединению данных: Компонента мета-уровня

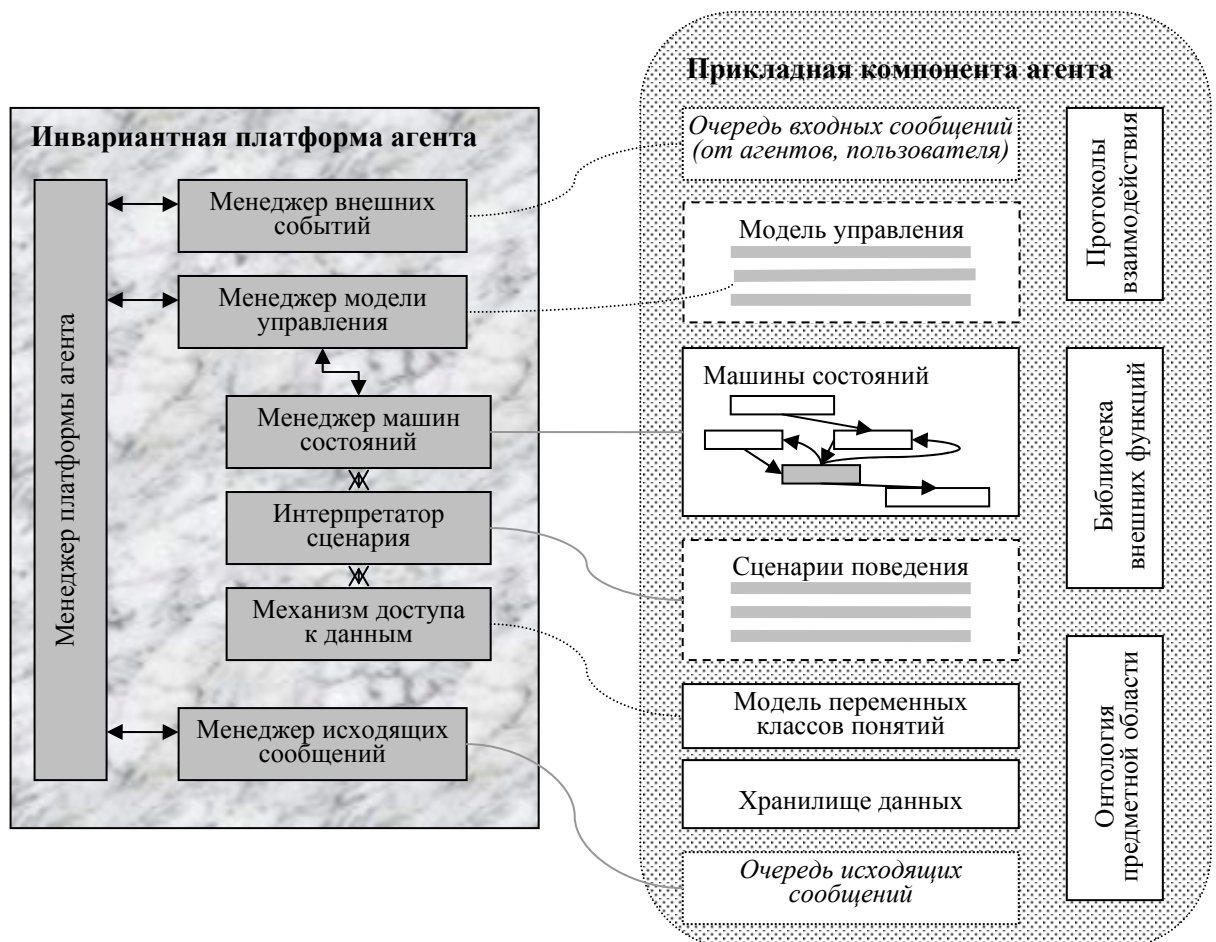


Рис.3. Структура *Generic Agent* и структура специфических знаний агентов

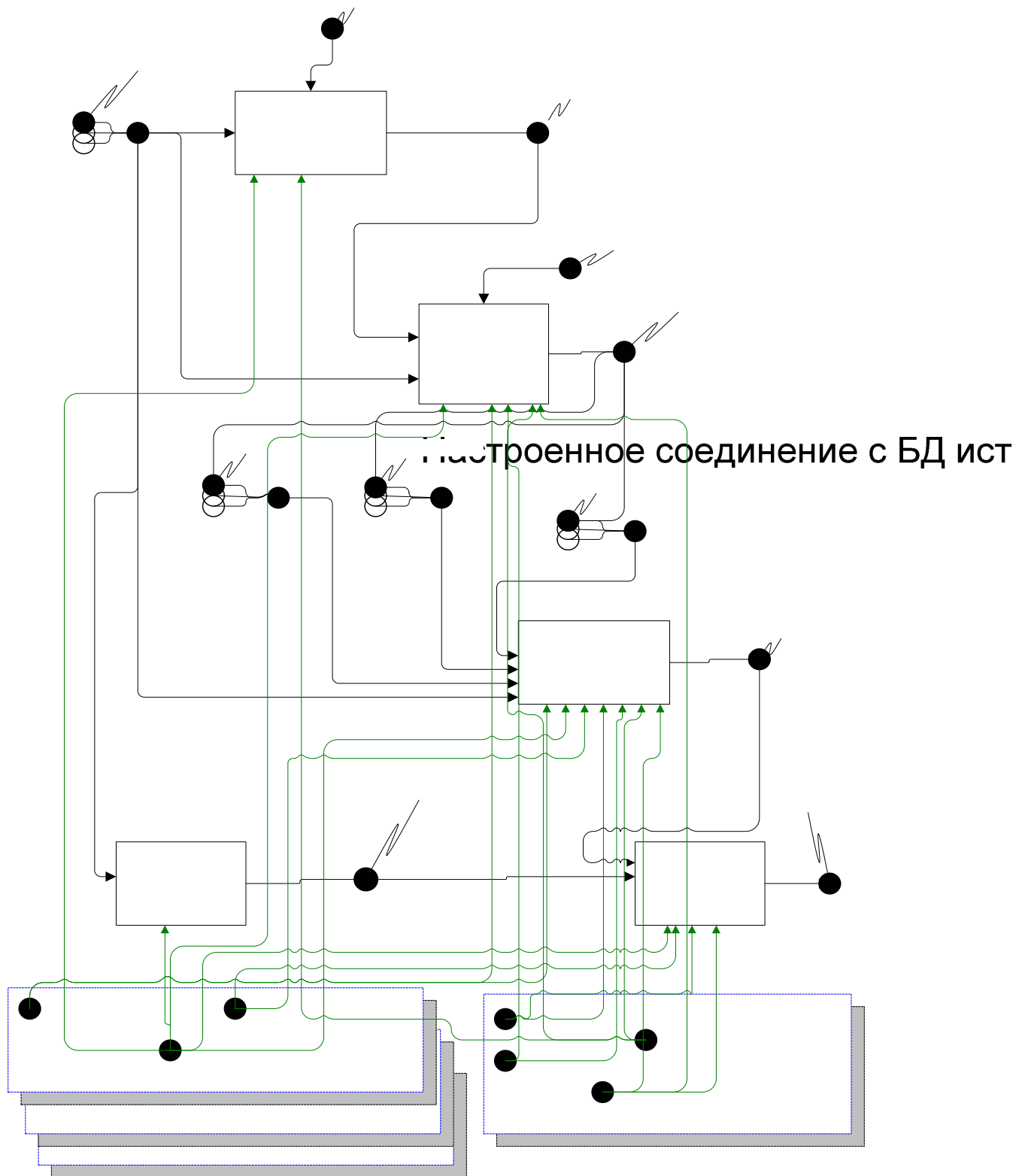


Рис.4. Протокол верхнего уровня взаимодействия агентов инструментальной системы при разработке приложения (формировании распределенной онтологии, распределенном обучении и формировании структуры принятия решения).

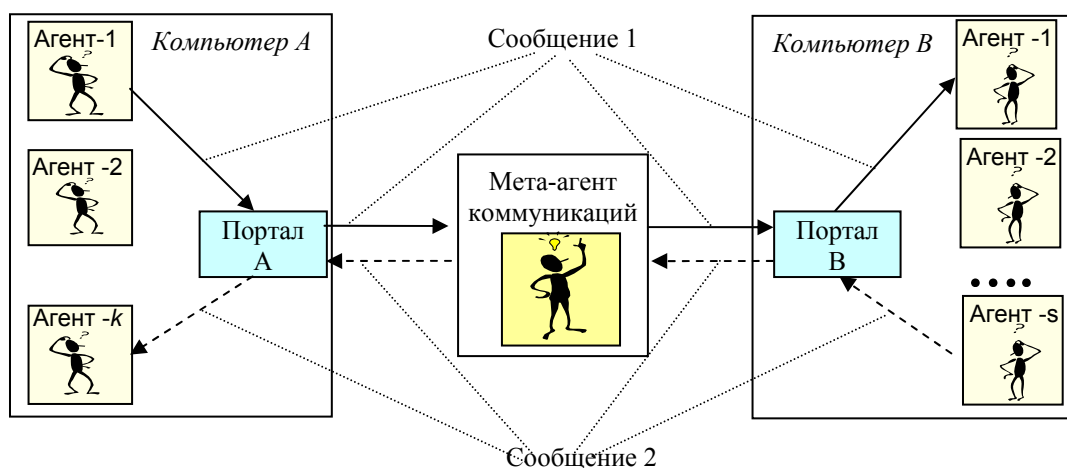


Рис.5 Схема обмена сообщениями между агентами

## Приложение 1.2. Другая дополнительная информация к основному тексту

*Краткое содержание Промежуточного отчета №3, представленного партнеру*

Предисловие

Глава 1. Типовые подзадачи задачи обучения объединению данных и их распределение между типовыми классами агентов системы обучения

- 1.1. Архитектура инструментального средства поддержки технологии разработки систем слияния данных
- 1.2. Типовые подзадачи задачи обучения объединению данных и их распределение между типовыми классами агентов системы обучения
- 1.3. Модели, языки и протоколы взаимодействия агентов и коммуникационная компонента

Глава 2. Архитектуры типовых агентов системы обучения, реализующих выбранные и разработанные алгоритмы извлечения знаний из данных

- 2.1. Введение
- 2.2. Типовой агент
- 2.3. Архитектура агента
- 2.4. Краткое описание технологии разработки прикладного программного агента

Глава 3. Мета-модель взаимодействия агентов в процессе распределенного обучения и принятия решений. Методы объединения решений

- 3.1. Введение
- 3.2. Мета-модель слияния данных
- 3.3. Технология разработки и протоколы верхних уровней, специфицирующие взаимодействие агентов инструментального средства
- 3.4. Разработка распределенной онтологии: концептуальное обоснование
- 3.5. Протоколы для распределенной разработки онтологии
- 3.6. Протокол взаимодействия агентов при разработке мета-модели объединения решений
- 3.7. Протокол взаимодействия агентов при принятии решений
- 3.8. Протокол распределенного обучения
- 3.9. Методы, используемые для объединения решений базовых классификаторов

Глава 4. Объектно-ориентированный проект инструментального средства поддержки технологии разработки систем слияния данных

- 4.1. Введение
- 4.2. *Uses cases* диаграммы
- 4.3. Другие компоненты объектно-ориентированного проекта

Заключение по отчету

### Приложение 1.3. Резюме статей и докладов, опубликованных за рассматриваемый год

1. Промежуточный отчет № 3. Декабрь 2002.
2. V.Gorodetski, O.Karsaev, I.Kotenko, A.Khabalov. Software Development Kit for Multi-agent Systems Design and Implementation. In B.Dunin-Keplicz, E.Navareski (Eds.), From Theory to Practice in Multi-agent Systems. *Lecture Notes in Artificial Intelligence*, Vol. # 2296, pp.121-130, 2002.

**Абстракт.** Статья представляет разработанную технологию и инструментальное средство для разработки и программной реализации многоагентных интеллектуальных систем. Инструментальное средство состоит из двух компонент: "Типового агента" и "Набора средств разработки многоагентной системы". Первая компонента состоит их программных классов, написанных на языке Visual C++ и Java, а также типовых структур данных и знаний. Вторая компонента включает в себя набор пользовательских средств-редакторов, снабженных "дружественными" пользовательскими интерфейсами, предназначенными для описания приложений и их инсталляции (развертывания) в среде конкретной компьютерной сети. Разработанное инструментальное средство использовано для быстрого прототипирования многоагентных систем системы обнаружения вторжений в компьютерные сети и моделирования атак на компьютерные сети.

3. V.Gorodetski, O.Karsaev, V.Samoilov. Data Fusion and Semantic Web: Meta-Models of Distributed Data and Decision Fusion. Proceedings of the International Workshop Semantic Web Mining. Organized within 6th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'02), Helsinki, August 19-23, 2002.

**Абстракт.** В соответствии с проектом, финансируемым Европейским офисом аэрокосмических исследований и разработок, разрабатывается математическая модель, многоагентная архитектура и технология разработки систем слияния данных в различных приложениях. Все множество задач, которое должно решаться в рамках рассматриваемой проблемы, может быть разделено на две группы. Задачи первой группы являются общими для многих приложений из области искусственного интеллекта, а потому используемые здесь методы заимствуются из смежных областей, например, из машинного обучения, объектно-ориентированного проектирования, статистики и т.п. Вторая группа включает в себя задачи, специфичные именно для рассматриваемого класса приложений, а потому они требуют разработки специфичных методов решения. Часть этих задач имеет непосредственное отношение к проблематике, попадающей в область интересов программы Semantic Web. Хотя проект предполагает решение задач обоих классов, в докладе, в основном, рассматриваются только задачи первого типа.

Основные вопросы, рассматриваемые в докладе, касаются следующего:

1. Разработка мета-модели данных распределенных источников, основанной на использовании онтологии, и анализ возникающих при этом специфических задач.
2. Распределенное обучение и мета-модель комбинирования решений.
3. Многоагентная архитектура системы слияния данных и ее роль в различных приложениях, рассматриваемых в рамках программы Semantic Web.

4. V.Gorodetski. Multi-agent Data Fusion: Design and Implementation Issues. 5th International Conference on Information Fusion (Fusion-2002), CD Proceedings of the section "AFOSR Information Fusion Initiative". Annapolis, MD, USA, July 3-6, 2002. Abstract

**Абстракт.** Нижеследующие аспекты технологии разработки и программной реализации многоагентных систем слияния данных составляют содержание работы:

- (1) Каким образом может быть разрешена проблема слияния решений, полученных на основе распределенных источников, содержащих гетерогенные данные? В частности, каким образом может быть обеспечено однозначное понимание сообщений, которыми обмениваются агенты многоагентной системы слияния данных? Как может быть решена проблема идентификации данных об одном и том же объекте, представленном фрагментами в различных базах данных? Каким образом можно справиться с разнообразием шкал измерения и структур представления данных в различных источниках? В рамках основного вопроса возникает также ряд других частных задач.
- (2) Каким образом решается задача распределенного обучения, в частности, каким образом организуется слияние данных, полученных на основе локальных источников данных, использующих различные данные и методы классификации?
- (3) Какова архитектура системы слияния данных и каким образом целесообразно распределить различные задачи между ее компонентами?

В работе также рассматривается предложенная технология поддержки разработки многоагентных приложений, которая, в частности, используется для разработки многоагентных

систем слияния данных. Даются сведения о приложениях, использованных для отработки предложенных подходов и технологии в целом.

5. V.Gorodetski, O.Karsayev and V.Samoilov. Multi-agent Data Fusion Systems: Design and Implementation Issues. Proceedings of the 10th International Conference on Telecommunication Systems - Modeling and Analysis, Monterey, CA, October 3-6, vol.2, pp.762-774, 2002.

**Абстракт.** Рассматривается задача принятия решений на основе совместной обработки данных, полученных из распределенных источников об одном и том же объекте. Основное внимание уделяется технологическим аспектам разработки и программной реализации систем решающих задачи названного класса. В частности, в работе представлен анализ специфических проблем, возникающих при разработке таких систем, методы их решения, а также многоагентная архитектура системы распределенного обучения, имеющего целью построение систем объединения данных. Отдельно проанализированы особенности проблемы распределенного обучения, а также известные подходы к решению ее центральной задачи – обучение объединению решений классификаторов, принимающих решения на основе данных отдельных источников.

6. V.Gorodetski, O.Karsayev. Mining of Data with Missing Values: A Lattice-based Approach. Accepted for presentation in Workshop on the Foundation of Data Mining and Discovery in the 2002 IEEE International Conference on Data Mining, Maebashi TERRSA, Maebashi City, Japan December 9 - 12, 2002

**Абстракт.** Работа представляет теоретически строго обоснованный алгоритм извлечения правил из данных, в частности, содержащих пропуски значений. Концептуально этот подход близок к хорошо известному методу AQ и отличается от него алгоритмом поиска максимально обобщенных правил. Заметим, что эта часть алгоритма является ключевой, именно она определяет эффективность любого подобного алгоритма. Предложенный алгоритм основан на использовании результатов теории решеток, что позволяет существенно снизить затраты процессорного времени на поиск правил. Центральным моментом работы является использование разработанного алгоритма для нахождения точной верхней и точной нижней граней для семейств множеств максимально обобщенных правил, которые могут быть получены при всевозможных означиваниях пропущенных значений. Хотя предложенный алгоритм рассмотрен только для бинарных данных, он легко обобщается также и на другие типы данных, в частности, на категориальные, упорядоченные и числовые данные.

Руководитель исследований по задаче 1  
доктор технических наук профессор  
В.И.Городецкий