

# ОСНОВНЫЕ ПРИНЦИПЫ ОРГАНИЗАЦИИ СИСТЕМ БЫСТРОЙ ИНТЕГРАЦИИ ЗНАНИЙ

А.В. Смирнов, М.П. Пашкин, Т.В. Левашова, Н.Г. Шилов

Санкт-Петербургский институт информатики и автоматизации РАН

199178, Россия, Санкт-Петербург, 14 линия, 39

smir@iiias.spb.su

---

УДК 681.3

А. В. Смирнов, М. П. Пашкин, Т. В. Левашова, Н. Г. Шилов. **Основные принципы организации систем быстрой интеграции знаний** // Труды СПИИРАН. Вып. 1, т. 2. — СПб.: СПИИРАН, 2002.

**Аннотация.** Знания — это важный ресурс для любого вида деятельности. Современные тенденции обуславливают необходимость существования для поддержки принятия решений глобальной информационной среды, включающей в себя конечных пользователей и обладающей слабосвязанными источниками знаний (эксперты, базы знаний, репозитории, электронные документы и т.д.). Огромное разнообразие инструментариев организации источников знаний сделало актуальной проблему организации систем для быстрой интеграции знаний из распределённых источников инфосферы. Принципам организации и свойствам систем, занимающихся вопросом быстрой интеграции знаний из распределённых источников, и посвящена настоящая статья. — Библ.18 назв.

UDC 681.3

A. V. Smirnov, M. P. Pashkin, T. V. Levashova, N. G. Chilov. **Main organisational principles of the rapid knowledge fusion systems** // SPIIRAS Proceedings. Issue 1, v. 2. — SPb.: SPIIRAS, 2002.

**Abstract.** Knowledge is a critical resource for any activity. Current trends require using a global information environment, including end-users/customers and loosely coupled knowledge sources/resources (experts, knowledge bases, repositories, documents, etc.), for decision-making. The vast diversity of the knowledge source management tools has made the problem of rapid knowledge fusion from infosphere's distributed knowledge sources, actual. The paper is devoted to organisation principles and properties of the systems dealing with rapid knowledge fusion from distributed knowledge sources. — Bibl.18 items.

## 1. Введение

Бурное развитие Internet привело к тому, что всё большее количество информации по различным проблемным областям становится доступным пользователям. Ввиду большого количества инструментариев для предоставления информации актуальными стали проблемы совместимости различных форматов, организации инструментариев для поиска, распознавания и интеграции знаний из распределённых источников.

Большинство разработок по организации хранения знаний в Интернет основывается на форматах XML, RDF (например, система WebKB, язык DAML) [1–3]. Использование таких стандартов уменьшает время поиска необходимой информации и позволяет эффективно использовать доступные знания. Однако область стандартизации форматов представления информации лежит за пределами интересов данной статьи.

Современные тенденции обуславливают необходимость существования для поддержки принятия решений глобальной информационной среды (называемой далее "инфосфера"), включающей в себя конечных пользователей и обладающей слабосвязанными (loosely coupled) источниками знаний (эксперты, базы знаний, репозитории, электронные документы и т.д.). Инфосфера должна

быть чувствительной к запросам пользователя, открытой и гибкой. Инфосферу с выше перечисленным набором свойств будем называть "масштабируемой инфосферой".

Знания это важный ресурс для любого вида деятельности. Современные требования к обладающим ресурсами знаниям системам включают следующее.

- Скорость реагирования на изменения (Velocity). Система постоянно стремится находить пути уменьшения и/или компенсации вариабельности в запросах заказчика/потребителя и поставщиках/источниках знаний.
- Гибкость (Flexibility). Система должна быть готова к внезапным изменениям в целеполагании функционирования. Она обеспечивает свою гибкость, храня минимальный объем информации в источниках знаний.
- Настраиваемость на потребителя (Customizability). Система должна быть готова сформировать любую возможную конфигурацию проблемной модели (знаний), которую заказчик/потребитель может запросить.

Изначально, в процессе развития информационных технологий и вычислительной техники преобладали собственно вычислительные задачи, но постепенно всё большее значение стала приобретать работа с данными и коммуникациями. Развивалось направление слияния данных и управления данными. Со временем, этот процесс естественным образом начал приводить к возникновению и развитию технологий, поддерживающих управление и интеграцию знаний. Попытки соединить накопленные знания, в том числе разнородные, с целью их повторного использования, то есть интегрировать знания, привели к необходимости разработки методов, направленных на анализ содержимого знаний. Один из способов представления знаний, для которого применим интеллектуальный анализ, предлагают теории онтологий — смысловые теории о разновидностях, свойствах объектов и связях между ними. Для реализации операций по работе с онтологиями предлагается использовать многоагентную технологию.

В результате сформировалась необходимость разработать подход для быстрой интеграции знаний с целью глобального понимания происходящих процессов и явлений, динамического планирования и глобального обмена знаниями [4]. Под *интеграцией знаний* понимается процедура синтеза существующих знаний с целью получения нового. Настоящая статья посвящена принципам организации и свойствам подобных систем.

## **2. Общие принципы организации систем интеграции знаний в масштабируемой инфосфере**

В настоящее время существует ряд систем, ориентированных на решение подобных проблем [5–10]. К основным их функциям относятся:

- изучение проблемной области и формирование набора онтологий, описывающих проблемную область, в которой они используются;
- предоставление интерфейса конечному пользователю — потребителю знаний для ввода запроса на поиск знаний;
- поиск источников знаний;
- выбор, объединение и проверка знаний из распределённых источников;
- предоставление итогов поиска пользователю;
- сохранение результатов работы.

Процесс интеграции знаний включает в себя перечисленные ниже операции [11] и может быть представлен в виде схемы, показанной на рис. 1:

- получение знания из источников знаний и перевод его в форму, подходящую для последующего использования:
  - приобретение знания из внешних источников;
  - выбора знания из внутренних источников (локальных баз знаний);
- генерация знания: создание знания посредством обнаружения или вывода из существующего;
- внутренняя обработка знания: изменение знания системы посредством хранения приобретенного, выбранного или сгенерированного знания;
- представление знаний в заданном виде с целью дальнейшего использования во внешней среде в качестве результата работы системы;
- управление интеграцией знаний: планирование, координация и контроль операций, составляющих процедуру интеграции знаний.

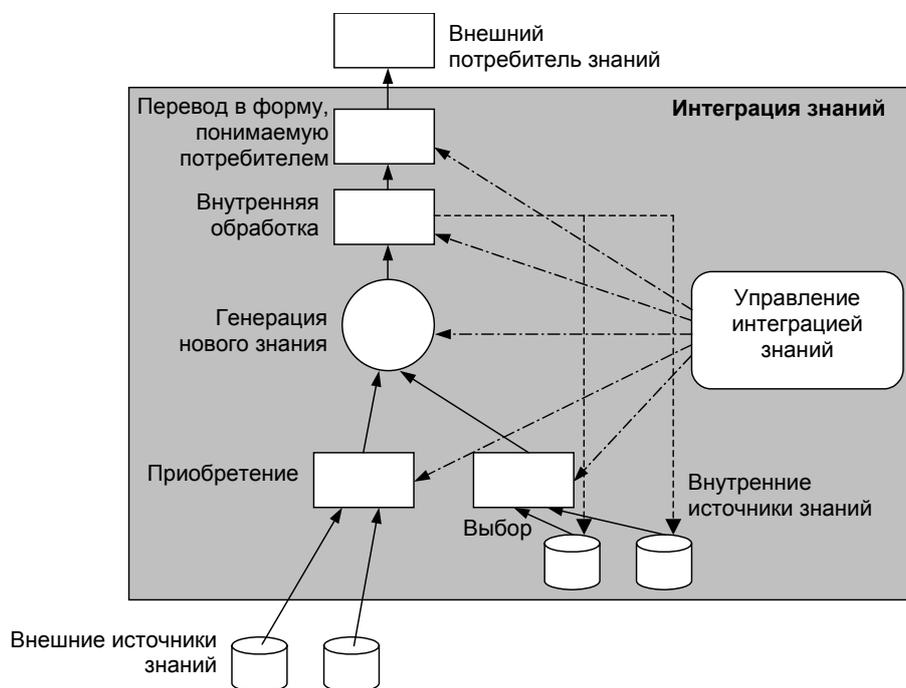


Рис. 1. Процесс интеграции знаний

Универсальный способ представления знаний, общий вид, к которому сводятся частные способы представления знаний, предлагают онтологии. Описанный процесс интеграции знаний сводится к построению онтологий и дальнейшей работе с ними, как с платформой для интеграции. Процесс интеграции знаний в терминах операций над онтологиями [11] представлен в табл. 1.

Таблица 1. Соответствие операций по интеграции знаний и операций над онтологиями

Операции интеграции знаний	Операции над онтологиями
<i>Приобретение (Acquisition)</i>	
Поиск подходящего знания из внешних источников посредством поиска, доступа, оценки и/или фильтрации	1. Поиск (finding)

Получение найденного знания посредством извлечения, накопления и/или сбора знаний, считающихся надежными, релевантными и важными	2. Перевод терминов (term translation) 3. Выполнение запроса к ресурсу (query) 4. Извлечение (extraction)
Организация полученного знания посредством очистки, ориентирования, интерпретации, компоновки и перевода в представление, пригодное для работы с ним.	5. Сравнение & перевод (matching & translation) 6. Выборка ИЛИ отсечение (slicing OR pruning)
Перенос организованного знания	
<i>Выбор (Selection)</i>	
Поиск подходящего знания во внутренних ресурсах системы посредством поиска, доступа, оценки и/или фильтрации	1. Поиск (finding)
Получение найденного знания посредством извлечения, накопления и/или сбора знаний, считающихся надежными, релевантными и важными	2. Выполнение запроса (query) 3. Извлечение (extraction)
Организация полученного знания посредством очистки и ориентирования	4. Выборка ИЛИ отсечение (slicing OR pruning)
Перенос организованного знания	
<i>Генерация (Generation)</i>	
Производство знания на базе существующего знания посредством создания, синтеза и построения	1. Создание новой онтологии (creating) 2. Соединение (merging) 2.1. Разбивка на модули (modularisation) 2.2. Пересечение (intersection) 2.3. Объединение (union) 2.4. Выборка ИЛИ отсечение (slicing OR pruning) 2.5. Сохранение (saving) 3. Проверка на непротиворечивость и полноту (validation) 4. Сохранение (saving)
Передача произведенного знания для внутренней обработки и/или подготовки для использования внешними пользователями	
<i>Внутренняя обработка (Internalization)</i>	
Оценка с необходимой очисткой и фильтрацией	1. Выборка ИЛИ Отсечение (pruning)
Доставка знаний к заданным ресурсам	2. Расширение (extension) 3. Обновление (update) <sup>1</sup>
<i>Подготовка для использования внешними пользователями (Externalization)</i>	1. Просмотр (browsing) 2. Перевод (translation) 3. Выпуск (publishing)

Ниже приведена таблица (табл. 2), сравнивающая некоторые характеристики трёх наиболее известных систем, ориентированных на интеграцию знаний.

Таблица 2. Сравнение характеристик существующих систем интеграции информации

Характеристика	KRAFT	OBSERVER	InfoSleuth
<b>Основные сведения</b>			
Разработчики	Universities of Aberdeen, Cardiff and Liverpool при поддержке British Telecom	University of Zaragoza, Spain	Microelectronics and Computer Technology Corporation (MCC),

<sup>1</sup> Операция не определена. Требуется дополнительное исследование и построение моделей и методов для ее реализации

Характеристика	KRAFT	OBSERVER	InfoSleuth
			Austin, Texas.
Цели проекта	Создать многоагентную систему для поиска информации с учётом ограничений	Организовать систему поиска информации независимо от структур и семантики данных в огромном числе репозиториях при помощи построения синонимичных отношений между онтологиями.	Создать открытую многоагентную систему для поддержки поиска информации в изменяющихся по содержанию и структуре источниках
Текущее состояние	Стартовал в мае 1996 г. Частично реализован.	Реализован прототип системы.	InfoSleuth закончен в июне 1997. В настоящее время ведётся проект InfoSleuth II. Есть реальные приложения.
Структура, инструментарий исполнения	KQML P/FDM, CoLan	Description Logics (DL) CLASSIC	KQML, KIF затем OKBC ODBC, затем JDBC LISP, CLISP, LDL+ Java, C/C++, NetScape
Ресурсы	Любые доступные информационные ресурсы, для которых подготовлены соответствующие механизмы обработки	Реляционные и объектно-ориентированные базы данных, HTML страницы	Изначально базы данных. В настоящее время любые доступные источники, для которых подготовлены соответствующие механизмы обработки
Доступ с мобильных устройств (WAP,...)	Нет	Нет	Предусмотрен
Экспериментальная область применения	Виртуальные предприятия	Библиографическая информация	Проект Environmental Data Exchange Network (EDEN)
Итерация поиска ин-	Нет	Да	Нет

Характеристика	KRAFT	OBSERVER	InfoSleuth
формации			
Хранение информации о пользователе	Неизвестно	Нет	Хранится история запросов, существуют механизмы предоставления её другим агентам и пользователям
<b>Работа с онтологиями</b>			
Базовые онтологии	Словарь WordNet, [http://www.cogsci.princeton.edu/~wn/]	Зависят от области применения.	Зависят от области применения
Выбор пользователем онтологий для работы	Нет	Да	Да
Отношения между онтологиями	Иерархия онтологий	Сеть онтологий	Отображение онтологии ресурсов в онтологии системы.
Пополняемость онтологий	Да	Да	Да.
Язык работы с онтологиями	P/FDM	DL	Сначала KIF и LDL+ Сейчас ОКВС
<b>Работа с агентами</b>			
Наличие агентов	Да	Нет	Да
Отношения между агентами	Специально выделенный агент Facilitator		Специально выделенный агент Broker
Базовые агенты	Интерфейсный агент (Wrapper), Маршрутизатор (Facilitators), Посредник (Mediators), Пользовательский агент (User Agent)		Агент ресурса (Resource Agent), Брокер (Broker Agent), Исполняющий агент (TaskExecution Agent), Пользовательский агент (User Agent)
<b>Дополнительные свойства</b>			
Доступ к БД	Зависит от источника	SQL	JDBC, SQL
Особен-	Работает с данными и с ограничениями.	Использование	Разработана

Характеристика	KRAFT	OBSERVER	InfoSleuth
ности		синонимов понижает избыточность. Изменение структуры репозитория приводит к изменениям в базовых онтологиях.	сеть взаимодействующих агентов, описаны механизмы обмена сообщениями открытых многоагентных систем

### 3. Особенности систем быстрой интеграции знаний

Учитывая накопленный научный опыт разработки аналогичных систем, и изучив рекомендации по созданию масштабируемых систем по сбору и анализу информации [13], разработаны принципы построения и работы систем быстрой интеграции знаний. В табл. 3 указаны специальные компоненты системы, обеспечивающие быструю интеграцию знаний в масштабируемой инфосфере.

Таблица 3. Компоненты системы быстрой интеграции знаний

Компонента системы	Описание компоненты
Агент пользователя	Агент для распознавания запроса пользователя и передачи полученных знаний пользователю
Макроагент интеграции знаний	Формирует новые знания на основе интеграции знаний, полученных из различных источников
Внутренняя база знаний	Внутренняя пополняемая база знаний. Служит для хранения и проверки знаний, возникших в результате интеграции знаний из доступных источников.
Картограммы знаний	Карты распределения знаний. Специальным образом организованное представление информации об источниках знаний, используемых для решения проблем.
Онтология приложения	Концептуальная модель, которая описывает область задачи реального мира, зависящая от конкретной области и проблемы
Онтология источника знаний	Описание источника знаний в терминологии онтологии приложения.
Онтология запроса	Результат перевода запроса пользователя в терминологию онтологии приложения
Планировщик поиска источников знаний	Инструмент, который на основе сформированной картограммы знаний и установленных ограничений оптимизирует и планирует процесс поиска знаний.
Пользовательский профайл (user profile)	Хранилище информации о пользователе, истории его обращений и т.д. (см. рис. 2). Составляющая доступна для использования в различных целях, в частности для ускорения поиска путём изучения истории обращений.
Средства диагностики	Встроенный инструмент для проверки информационных ресурсов на предмет доступности, изменений и корректности извлечённых знаний.

Основные принципы, закладываемые в систему, следующие.

1. Алгоритмы и методы функционирования системы разрабатываются независимо от области применения.
2. Система должна функционировать в конкретной проблемной области, изначально изученной и описанной экспертами. По итогам изучения должна быть создана онтология высокого уровня (будем называть её «базовая онтология») — структурированное представление знаний в терминологии проблем-

ной области и набор онтологий доступных информационных ресурсов (в терминах и понятиях базовой онтологии). Под информационными ресурсами в рамках системы понимаются возможные источники знаний. В процессе функционирования должно формироваться дерево онтологий информационной системы, в корне которого находится онтология высокого уровня, а на листьях — онтологии источников знаний. В узлах данного дерева будут располагаться онтологии задач — результат перевода запроса пользователя в терминологию онтологии верхнего уровня. Такая структура предполагает масштабирование дерева, позволяет формировать шаблоны пользовательских запросов для быстрого поиска знаний, облегчает повторное использование результатов запросов, ускоряет интеграцию знаний от онтологий ресурсов к базовой онтологии и выдачу результата запроса пользователю.

3. Система должна предоставлять интерфейс ввода запроса к знаниям, включающий в себя:
  - описание проблемы — формализацию с набором ограничений (временные, стоимостные и т.д.). Для ускорения и облегчения ввода запросов предусмотрена возможность использовать шаблоны запросов;
  - параметры поиска — online или offline;
  - форму получения результата — online, e-mail и т.п.
4. Система должна осуществлять трансляцию введённого запроса в свою базовую онтологию, производить декомпозицию запроса на составляющие — подзапросы, распознавать полученные подзапросы и передавать их на выполнение. Под выполнением подзапросов понимается определение системой источников знаний, которые дают ответ по проблеме, заданной пользователем, обращение к ним, фильтрация в соответствии с ограничениями пользователя, интеграция знаний из различных источников, проверка на непротиворечивость, проверка на удовлетворение заданным ограничениям, предоставление результатов пользователю. Подзапросы выполняются параллельно. Благодаря древовидной структуре, формирующейся на основе картограммы знаний, определяются веса для источников знаний, и оптимизируется маршрут доступа к ним.
5. Если в результате поиска знания обнаружены не будут, то при наличии доступных экспертов, специализирующихся в данной проблемной области, и согласии пользователя на работу с ними, запрос может передаваться на рассмотрение экспертов. В случае предоставления экспертом знаний по запросу внутренняя база знаний может пополниться и соответственно расширится и картограмма знаний.
6. Результаты работы должны протоколироваться. Заносятся изменения в пользовательский профайл, при необходимости, во внутреннюю базу знаний и картограмму знаний с целью повторного использования в будущем для решения аналогичных задач.

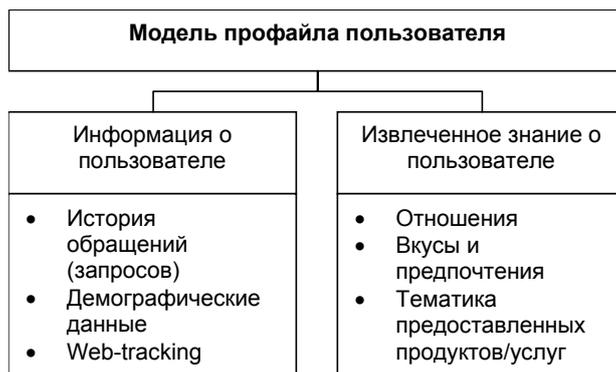


Рис. 2. Структура профиля пользователя

Выделим следующие возможные источники знаний.

- Внутренняя база знаний системы, аккумулирующая генерируемые новые знания при помощи встроенных средств.
- Эксперты, вводящие знания напрямую по запросу пользователя при помощи встроенных средств.
- Доступные внешние базы данных. Из них извлекается структура таблиц, определяются отношения между ними и диапазоны предельных значений.
- Доступные внешние базы знаний. Путём трансляции используемых онтологий между системой и источниками извлекается необходимые знания.
- Структурированные документы (текстовые и HTML документы). Путём поиска информации по ключевым словам определяется соответствие документа запросу.
- Прочие источники. По мере развития систем число источников, с которыми система сможет оперировать, будет возрастать. С этой целью, для каждого типа источника будут разрабатываться свои механизмы распознавания и извлечения знаний (рис. 3).

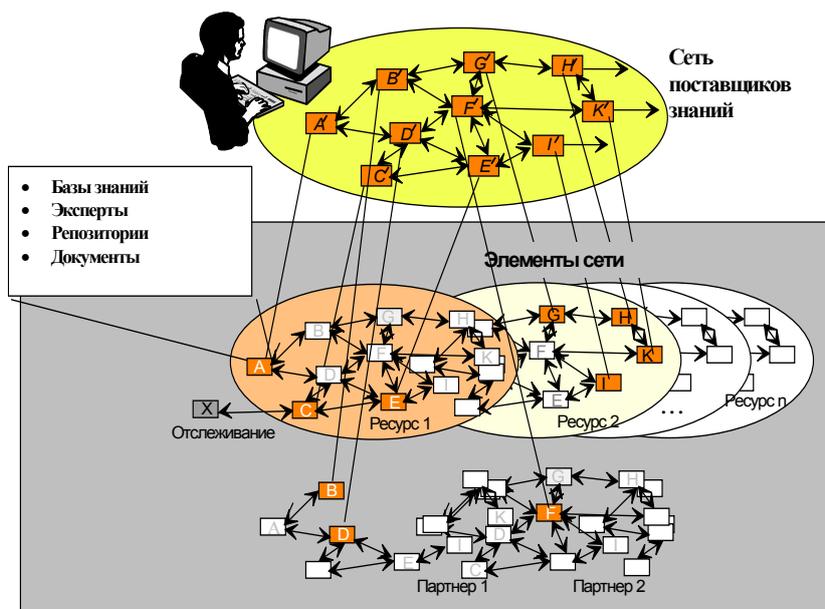


Рис. 3. Генерация поиска альтернативных ресурсов

Выделим следующие технологии, необходимые для реализации среды интеграции знаний [14].

- Прямая запись знаний специалистами по проблемной области (основанная на использовании графического интерфейса пользователя и объектно-ориентированной библиотеке типовых структур).
- Параллельная разработка репозитория знаний распределенной группой экспертов/специалистов (основанная на автоматическом распространении изменений, клонировании моделей сложных объектов и технологии переговоров при разрешении конфликтов).
- Организация совместно используемых знаний на основе создания картограмм (карт распределения) знаний (основанная на теории повторно используемых онтологий и фрагментов знаний).
- Распределенное управление нечеткими знаниями (основанное на модели объектно-ориентированных динамических сетей ограничений как общей парадигме используемых онтологий).

#### 4. Прототип среды «Ассистент» для работы с онтологиями

В качестве инструмента для описания информационных ресурсов и извлечения знаний о ресурсе рассмотрим прототип системы «Ассистент», разработанной в лаборатории интегрированных систем автоматизации СПИИРАН [15–17].

Система «Ассистент» ориентированна на построение онтологий с использованием баз данных для хранения информации. Доступ к данным осуществляется при помощи средств DAO/ODBC. В качестве примера для использования предлагается проблемная область «Реинжиниринг бизнес процессов». Соответствие между понятиями, используемыми в системе, и терминологией, используемой в формате представления знаний KIF (Knowledge Interchange Format) [17], приведено в табл. 4.

Основными компонентами системы являются:

- 1) редактор типов (Type Editor) — средство формирования и модифицирования онтологий;
- 2) редактор иерархии объектов (Hierarchy) — средство формирования и модифицирования иерархических связей между объектами онтологии.

В зависимости от специфики проблемной области система может быть дополнена другими компонентами, предназначенными для решения задач конечного пользователя на основе информации, содержащейся в основной базе. В настоящее время разработаны средства для планирования производственной программы предприятия (Goal), проектирования производственных комплексов (Module), а также для управления производственными ресурсами (Goal&Module).

Выполнение системы в многопользовательском варианте предполагает возможности отдельного выполнения отдельных стадий реинжиниринга и ввода информации различными специалистами на своих рабочих местах, объединенных локальной сетью.

Прототип системы построен таким образом, что доступ к данным может быть легко осуществлен из различных приложений (например, MS Excel), что значительно расширяет возможности ввода и использования информации, хранящейся в основной БЗ.

Таблица 4 Соответствие понятий, используемых в системе «Ассистент», и понятий KIF

Понятие в системе «Ассистент»	Термин KIF	Определение
Class (класс)	Primitive Class	Значения шаблонного (наследуемого подклассами) слота не определяют набора необходимых и достаточных условий для отнесения элемента к экземплярам класса
Type (тип)	Non-Primitive Class	Значения шаблонного слота определяют набор необходимых и достаточных условий для отнесения элемента к экземпляру класса
Attribute (атрибут)	Template slot	Описывает значения собственного слота, содержащиеся во всех экземплярах класса
Object (объект)	Instance	Экземпляр класса

Редактор типов представляет собой средство для формирования, модифицирования и управления базой знаний.

Основными единицами базы данных системы «Ассистент» являются объекты (например, изделие, машина, операция и т.п.).

Наивысшим уровнем группировки объектов являются классы, которые отражают природу объектов. Классы определяются и заносятся в базу администратором на этапе инсталляции.

Следующим уровнем группировки объектов являются типы. Каждый тип принадлежит одному из классов системы и содержит объекты со сходными свойствами. Пользователь системы может создавать, редактировать и удалять типы объектов.

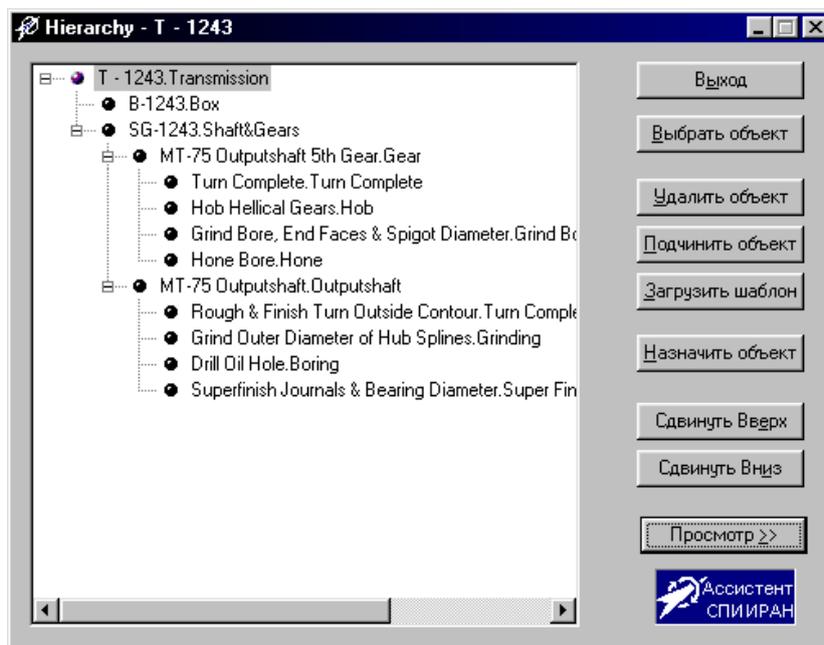


Рис. 4. Редактор иерархии объектов

На данном уровне также определяются атрибуты объектов (объекты одного типа имеют одинаковый набор атрибутов). Атрибуты могут быть двух категорий — статические (неизменяющиеся во времени) и исторические (изменяющиеся во времени) и четырех типов:

- целое число (Integer);
- число (Double);

- строка (String);
- дата (Date).

На уровне объектов атрибутам присваиваются значения.

Структура базы знаний позволяет устанавливать отношения между классами и типами объектов (например, операция типа «Сверление» может быть выполнена на станках типов «Сверлильные» и «Токарные»). Кроме того, на уровне типов реализована возможность задания сложных систем ограничений на значения атрибутов. Каждое ограничение может состоять из произвольного числа бинарных выражений, соединенных между собой логическими операторами AND или OR (например, длина и диаметр обрабатываемой заготовки не должны превосходить максимально допустимых для токарных станков и т.п.).

Редактор иерархии объектов представляет собой средство для формирования и модифицирования иерархических связей между объектами онтологий. Это могут быть: структуры составных объектов (рис. 4), последовательности операций для выпуска деталей, варианты размещения и т.п.

Редактор иерархии объектов поддерживает наследование подчиненных объектов, что допускает поэтапное создание сложных иерархических систем объектов, и использование шаблонов, автоматизирующих работу пользователя.

## 5. Заключение

На основе анализа существующих проектов в области интеграции информации разработана методология быстрой интеграции знаний из распределённых источников и спецификация требований к системам быстрой интеграции знаний в масштабируемой инфосфере, определены технологии интеграции знаний из распределённых источников. Выделены типы информационных ресурсов для систем быстрой интеграции знаний и описан фрагмент прототипа компьютерной среды. Данное исследование было выполнено в рамках проекта № 4.4 по программе Президиума РАН «Интеллектуальные компьютерные системы». Система «Ассистент» разработана в рамках проекта с Исследовательским центром компании Форд (Ахен, Германия).

## Литература

- [1] Karp P. D., Chaudhri V. K., Thomere J. F. XOL: An XML-Based Ontology Exchange Language. — Technical Note 559, AI Center, SRI International, July 1999.
- [2] Martin P., Eklund P. Embedding Knowledge in Web Documents // *Proceedings of the Eighth International World Wide Web Conference*, Toronto, May 11–14, 1999.
- [3] Hendler J., McGuinness D. L.. The DARPA Agent Markup Language // *IEEE Intelligent System*, 2000. — V. 15. — No. 6.
- [4] Tsui E., Garner B. J., Staab S. The role of artificial intelligence in knowledge management // *Knowledge-Based Systems*, 2000. — V. 13. — Pp. 235–239.
- [5] Mena E., Kashyap V., Sheth A., Illarramendi A. OBSERVER: An Approach for Query Processing in Global Information Systems based on Interoperation across Pre-existing Ontologies // *Distributed and Parallel Databases*, 2000. — V. 8. — No. 2. — Pp. 223–271.
- [6] Nodine M. H., Unruh A. Facilitating Open Communicating in Agent Systems: the InfoSleuth Infrastructure. Technical Report MCC-INSL-056-97. — Microelectronics and Computer Technology Corporation, Austin, Texas, 1997 — 78759.

- [7] Preece F., Hui K., Gray A., Marti P., Bench-Capon T., Jones D., Cui Z. The KRAFT architecture for knowledge fusion and transformation // *Knowledge-Based Systems*, 2000. — V. 13. — Pp. 113–120.
- [8] Aguirre J. L., Brena R., Cantu F. J. Multiagent-based knowledge networks // *Expert Systems with applications*, 2001. — V. 20. — Pp. 65–75.
- [9] Tamma V. A. M., Vissler P. R. S. Integration of Heterogeneous Sources: Towards a Framework for comparing Techniques // *Proceedings of the AI\*IA '98 Workshop on Techniques for Organization and Intelligent Access of Heterogeneous Information*, Padova, Italy, 1998. — Pp. 89–93.
- [10] Maikovich N. V. Multiagent system for information gathering and monitoring in the Internet // *Proceedings of the Second Conference on Management and Control of Production (MCPL)*. Grenoble, France, 2000.
- [11] Holsapple C. V., Singh M. The knowledge chain model: activities for competitiveness // *Expert System with Applications*, 2001. — V. 20. — Pp. 77–98.
- [12] Левашова Т. В. Принципы управления онтологиями, используемые в среде интеграции знаний // Труды СПИИРАН. Вып. 1, т. 2. — СПб.:СПИИРАН, 2002.
- [13] Bowman C. Mic, Danzig P. B., Manber U., Schwartz M. F. Scalable Internet Resource Discovery // *Research Problems and Approaches. Communications of the ACM*, 1994. — V. 37. — No. 8. — Pp. 98–107.
- [14] Smirnov A. V. Rapid Knowledge Fusion into the Scalable Infosphere: A Concept and Possible Manufacturing Applications // *Proceedings of the International NAISO Congress on Information Science Innovations (ISI'2001), Symposium on Intelligent Automated Manufacturing (IAM'2001)*, Dubai, U.A.E., 2001.
- [15] Chilov N., Smirnov A. Management Decision-Making Models And Tools For Estimating Strategic Objectives of Virtual Enterprise Participants // *Proceedings of the 1999 Advanced Summer Institute, 2nd International Workshop on Intelligent Manufacturing Systems IMS'99*. Leuven, Belgium, 1999. September 22–24.
- [16] Турбин П. А., Шилов Н. Г. Организация объектно-ориентированной среды информационной поддержки принятия решений “Ассистент” // *Информационные технологии и интеллектуальные методы*. СПб: СПИИРАН, 1999. — № 3.
- [17] Смирнов А. В., Данилов К. Д., Левашова Т. В., Шилов Н. Г. «Ассистент» — объектно-ориентированная среда для конфигурирования сложных объектов // *Тез. докл. конф. «Проблемы автоматизации, идеи, решения — 99»*, Севастополь, Россия, 1999.
- [18] FIPA 98. — Specification. Part 12. Ontology Service. — Geneva, Switzerland, Foundation for Intelligent Physical Agents (FIPA), 1998. — Version 1.0.