

Заявляемый приоритет

Данное изобретение заявляет приоритет ранее поданных предварительных патентных заявок США № 60/300385 от 22 июня 2001 г. и № 60/360610 от 28 февраля 2002 г.

Уведомление об авторском праве

Данное раскрытие защищено законами США и Международного авторского права. © 2002 Nosa Omoigui. Авторские права защищены. Часть раскрытия существа изобретения в данном патентном документе содержит материал, который является предметом защиты авторским правом. Владелец авторского права не имеет возражений против факсимильного воспроизведения кем-либо патентного документа или патентного раскрытия в том виде, как оно представлено в патентном фонде или реестре патентов Ведомства по патентам и товарным знакам, но в остальном сохраняет за собой все авторские права.

Область техники

Настоящее изобретение в целом относится к системам управления информацией и, более конкретно, к интегрированной или «бесшовной» (непрерывной) реализации базовой структуры и получаемой в результате среды для поиска, управления, доставки и представления знаний.

Предшествующий уровень техники

В настоящее время знания повсеместно признаются как основное имущество (актив) для организаций во всем мире и как инструмент для получения преимуществ в конкурентной борьбе. В современном связанном, основанном на информации мире, специалисты в области информационных технологий должны иметь доступ к знаниям и инструментальным средствам для принятия лучших, более быстрых, и более информированных решений, чтобы повышать свою производительность, расширять связи с заказчиками, и делать собственный бизнес более конкурентоспособным. Кроме того, промышленные эксперты рекламировали "быстроту" и "предпринимательство в реальном времени" в качестве важных целей коммерческой деятельности, присутствующих в информационной экономике.

Многие организации начали понимать значение распространения знания в рамках их деятельности для улучшения продукции и обслуживания заказчиков и значение использования хорошо обученной рабочей силы. Инвестиции, осуществляемые предприятиями в e-Learning (электронное обучение) и корпоративное обучение, могут служить свидетельством этому. Компании также инвестируют капитал в инструментальные средства, предназначенные для управления содержимым (информационным содержанием) поиска, сотрудничества и сбора коммерческой информации. Компании также расходуют значительные ресурсы на перевод в цифровую форму их бизнес-процессов, особенно касающихся приобретения и хранения заказчиков.

Однако многие активы, относящиеся к сферам знаний/обучения и связей с заказчиками по-прежнему сохраняются в разнотипном наборе хранилищ, не понимающих язык друг друга, и в результате ими управляют и взаимодействуют как с независимыми «островками» информации. В принципе, то, что многие организации считают "знаниями", является просто данными и информацией. Информационная экономика в значительной степени является борьбой за отыскание пути предоставления контекста, значения (смысла) и эффективного доступа к этому все возрастающему объему данных и информации или, иными словами, превращения объема доступных данных и информации в полезное знание.

Информация была долгое время доступна в различных формах, таких как газеты, книги, средства аудиовизуальной информации, радио и телевидение, и в электронной форме с изменяющимися степенями распространения. Управление информацией и доступ к ней сильно изменились с использованием компьютеров и вычислительных сетей. Связанные в сеть вычислительные системы обеспечивают доступ во всей системе к информации, поддерживаемой в любой точке в системе. Пользователям необходимо только установить необходимое соединение с сетью, обеспечить надлежащую авторизацию, и идентифицировать требуемую информацию, чтобы получить доступ.

Доступ к информации был дополнительно усовершенствован с появлением сети Интернет, которая соединяет большое количество компьютеров, имеющих различное географическое положение, чтобы обеспечить доступ к огромному объему информации. Наиболее широко распространенный способ предоставления информации по сети Интернет - через «Всемирную паутину» или Web. Web состоит из множества компьютеров или Web-серверов, соединенных с сетью Интернет, в которой обычно функционируют серверы протоколов HTTP (протокол передачи гипертекста), FTP (протокола передачи файлов), GOPHER или другие серверы. Web-серверы содержат Web-страницы на Web-сайтах. Web-страницы кодируются с использованием одного или более языков типа первоначального HTML (языка разметки гипертекста) или более современного XML (расширенного языка разметки) или SGML (стандартного обобщенного языка разметки). Изданные описания для этих языков включены в настоящий документ посредством ссылки. К Web-страницам, представленным на этих языках форматирования, пользователи Интернет могут осуществлять доступ через программное обеспечение для просмотра и навигации в сети, такое как Internet Explorer компании Microsoft или Navigator компании Netscape.

Сеть Web в значительной степени была сформирована на основе синтаксиса и структуры, а не на контексте и семантике. В результате доступ к информации обычно осуществляется через поисковые серверы и Web-каталоги. Современные поисковые серверы используют ключевое слово и соответствующие способы поиска, которые основаны на текстовой информации или информации по базовым темам и ин-

дексах без ассоциированной контекстной и семантической информации. К сожалению, такие способы поиска приводят к получению огромного количества результатов, в значительной степени не отвечающих запросу, - документов, в отличие от действенного знания. Были разработаны усовершенствованные способы поиска, чтобы «сфокусировать» запросы и повысить релевантность поисковых результатов. Многие из таких способов основаны на предшествующих тенденциях пользовательского поиска в формировании базовых допущений относительно требуемой информации. В качестве альтернативы, другие способы поиска основаны на классификации Web-сайтов для дополнительного «фокусирования» результатов поиска на областях, ожидаемых как наиболее релевантные. Независимо от способа поиска, основополагающая структура информации, по которой осуществляется поиск, является индексно-определяемой, а не контекстно-определяемой. Частота или тип текстовой информации, ассоциированной с документом, определяет результаты поиска в отличие от признаков (атрибутов) сущности (предмета) документа и того, как эти признаки связаны с пользовательским контекстом. Результатом является продолжающаяся сохраняться неоднозначность и неэффективность, свойственная использованию сети Web в качестве инструментального средства приобретения действенных знаний.

Сегодня на предприятиях во всем мире сеть Web является информационной платформой для специалистов в области информационных технологий, и в этом состоит проблема. Сеть Web, как известно, является платформой для данных и информации, тогда как ее пользователи действуют на уровне "знаний". Этот разрыв является фундаментальным и не должен недооцениваться. Сеть Web в значительной степени осуществила мечту об "информации на кончиках ваших пальцев". Однако специалисты в области информационных технологий требуют "знания на кончиках пальцев" в отличие от просто "информации на кончиках пальцев". К сожалению, современные специалисты в области информационных технологий используют сеть Web для просмотра и поиска документов - как компиляции данных и информации, а не фактических знаний, релевантных их запросу. Для получения улучшенного знания требуется обеспечение надлежащего контекста, значения и эффективного доступа к данным и информации, что отсутствует в традиционной сети Web.

Делались попытки достижения цели как получение "знания на кончиках пальцев". Одним примером является новая концепция организации и распространения информации, упоминаемая как семантическая сеть Web. Семантическая сеть Web является расширением существующей сети Web, в которой информации дано строго определенное значение, позволяющее компьютерам и пользователям лучше работать во взаимодействии. Хотя это является существенным шагом вперед в поддержке улучшенного контекста, значения и доступа к информации в Интернет, однако, семантическая сеть Web должна еще найти успешную реализацию, соответствующую ее потенциальным возможностям.

Как современная сеть Web, так и семантическая сеть Web не в состоянии обеспечить надлежащий контекст, значение и эффективный доступ к данным и информации для приобретения пользователем действенных знаний. Частично это проблема относится к способам структурирования современной сети Web и предложенной семантической сети Web или, другими словами, связана с их технологическими уровнями. Как показано на фиг. 1, современная сеть Web, например, которая является гипертекстовой средой, обеспечивает три технологических уровня, которые включают в себя «неинтеллектуальные» связи или связи без контекстной зависимости, временной зависимости и т.д. Современные принципы семантической сети Web, определяемой также как "семантическая гиперсреда", предусматривают пять технологических уровней, как показано на фиг. 2. Как объяснено более подробно ниже, есть серьезные ограничения, связанные со структурами каждого из технологических уровней.

Кроме того, во всесторонней системе управления информацией должны присутствовать различные свойства, чтобы обеспечить интегрированную и «бесшовную» (непрерывную) среду реализации базовой структуры и получаемую в результате среду для поиска, управления и доставки знаний. Не являющийся исчерпывающим перечень таких свойств включает следующее: Семантику/Значение; Контекстную зависимость; Временную зависимость; Автоматическую и интеллектуальную «Обнаружимость»; Динамическое связывание; Управляемые пользователем навигацию и просмотр; Участие в сети «не HTML»-документов и локальных документов; Гибкое представление, которое интеллектуально передает семантику отображаемой информации; Логика, Логический вывод и Умозаключение; Гибкий управляемый пользователем анализ информации; Гибкие семантические запросы; Поддержку считывания/записи; Аннотирование; "Сеть Web доверия"; Информационные пакеты (элементы сопряжения); Шаблоны контекста и ориентированное на пользователя агрегирование информации. Каждое из этих свойств будет обсуждено ниже в контексте их применения как к современной сети Web, так и к семантической сети Web.

Семантика/Значение

Современная сеть Web не имеет семантики как внутренней части платформы и пользовательского опыта. Web-страницы передают только текстовые и графические данные, а не семантику данных, которые они содержат. В результате пользователи не могут осуществлять семантические запросы типа таких, которые можно было бы ожидать от естественного языка, например, "найдите мне все книги, имеющие менее чем сто страниц, о джазе Латинской Америки и изданные за последние пять лет". Чтобы иметь возможность обработать такой вопрос, Web-сайт или поисковый сервер должен "знать" книги, которые он содержит, и должен иметь возможность «интеллектуально» отфильтровывать их содержание на осно-

вании семантики вопроса в запросе. Такой вопрос в сети Web в настоящее время не возможен. Вместо этого пользователи вынуждены полагаться на поиски, основанные на тексте. Такие поиски обычно приводят к информационной перегрузке или информационной потере, поскольку пользователь вынужден выбирать поисковые термины, которые могут не совпадать с текстом в информационной базе. В вышеупомянутом примере, пользователь мог бы выбрать поисковый термин "Книги Латинская Америка Джаз" и надеяться, что поисковый сервер сможет осуществить соединение. Затем обычно пользователю остается независимая фильтрация результатов поиска. Такой тип основанного на тексте поиска также включает в себя те термины, которые могли бы передавать то же самое значение. В вышеупомянутом примере результаты поиска терминов типа "Книги по джазу Южной или Центральной Америки" или "Публикации по джазу из стран Латинской Америки" могли бы игнорироваться в процессе обработки поискового запроса.

Отсутствие семантики также означает, что современная сеть Web не дает возможность пользователям осуществлять навигацию по ресурсам сети способом, на основании хода размышлений людей. Например, пользователю может потребоваться навигация по корпоративной сети интранет, используя организационную структуру предприятия. Например, от персонала к документам, которые они создают, к экспертам по этим документам, к непосредственным отчетам этих экспертов, к перечням распределения, элементами которых является непосредственные отчеты, к элементам перечня распределения к документам, которые образуют элементы, и т.д. Такая "сеть Web" является семантической и основана на классификации фактической информации ("сущности"), а не только "страниц", как в современной сети Web.

Отсутствие семантики также имеет другие следствия. Во-первых, это означает, что сеть Web не является программируемой. При наличии семантики сеть Web могла бы использоваться Интеллектуальными агентами, которые могут придать значение страницам и связям и затем делать логические выводы, рекомендации, и т.д. При современной сети Web, единственным "Агентом", который может делать логические выводы, является человеческий мозг. Как таковая, сеть Web не использует огромные вычислительные возможности, которыми обладают компьютеры, поскольку она (семантика) не представлена способом, который компьютеры могут понимать.

Отсутствие семантики также означает, что информация не является действенной. Поисковый сервер не "понимает" результаты, которые он выдает. По существу, как только пользователь принимает результаты поиска, он предоставлен сам себе. Также Web-браузер не "понимает" информацию, которую он отображает, и не может проводить интеллектуальных операций с информацией. При наличии семантики интеллектуальный дисплей, например, будет "знать", что событие является событием и могло бы делать нечто интересное, подобное проверке, находится ли событие уже в календаре пользователя, отображению информации свободен/занят, или предоставления возможности пользователю автоматически вставлять событие в свой календарь, таким образом делая информацию действенной. Информация, представленная без семантики, не является действенной или может требовать, чтобы семантика была логически выведена, что может приводить к нежелательному пользовательскому опыту.

Семантическая сеть Web пытается устранить недостатки, связанные с ограничениями семантики/значения в современной сети Web, посредством кодирования информации со строго определенной семантикой. Web-страницы в семантической сети Web включают в себя метаданные и семантические связи с другими метаданными, таким образом позволяя поисковым серверам выполнять более интеллектуальные и точные поиски. В дополнение семантическая сеть Web включает в себя онтологию, которая будет использоваться для представления знаний, таким образом позволяя серверу семантического поиска интерпретировать термины на основании значения, а не просто текста. Например, в предыдущем примере онтология понятия «Латиноамериканский джаз» могла бы использоваться на сайте семантической сети Web и позволила бы поисковому серверу на сайте "знать", что термины "Книги по джазу Южной или Центральной Америки" или "Публикации по джазу из стран Латинской Америки" имеют то же значение, что и термин "Книги по Латиноамериканскому джазу". Хотя концептуально преодолены многие из недостатков современной сети Web, однако, до настоящего времени отсутствовала успешная реализация строго определенной модели данных, обеспечивающей контекст и значение, включающие в себя, в частности, необходимые семантические связи, онтологию и т.д. для обеспечения дополнительных характеристик, таких как контекстная зависимость и временная зависимость.

Контекстная зависимость

Современная сеть Web не имеет контекстной зависимости. Следствием отсутствия контекста является то, что современная сеть не является персонализированной. Например, документы в доступном хранилище являются автономно статическими и поэтому «тупыми». Информация, релевантная предмету документа, уже была опубликована, публикуется вновь или будет скоро опубликована. Однако поскольку документ в хранилище является статическим, нет никакого способа динамически ассоциировать его предмет с этой релевантной информацией в реальном времени. Иначе говоря, пользователи не имеют способа для динамического соединения в реальном времени их частного контекста с внешней информацией. Источники информации (такие как документ), которые формируют контекст, «сидят» в своих собственных «островках», полностью изолированы от других релевантных источников информации. Это приводит к потерям производительности и информации.

Первопричиной этого является то, что современная сеть Web является ориентированной на представление средой, предназначенной для представления видимого изображения или вида информации «немому» («неинтеллектуальному») клиенту (например, удаленному компьютеру). Клиент фактически не играет никакой роли в пользовательском опыте, кроме простого отображения того, что сервер указывает отобразить. Даже в тех случаях, когда имеется код клиентской стороны (подобный Java-апплетам и ActiveX-компонентам управления), компоненты управления обычно выполняют некоторую конкретную функцию без координированного действия с удаленным сервером, так что клиентский код управляется серверным кодом.

С точки зрения производительности следствием является то, что специалисты в области информационных технологий и потребители информации находятся полностью во власти создателей информации. Сегодня специалисты в области информационных технологий имеют общедоступные узлы компьютерной сети (порталы), которые поддерживаются и обновляются для обеспечения настраиваемых представлений корпоративной информации, внешних данных и т.д. Однако это по-прежнему является ограничивающим, поскольку специалисты в области информационных технологий полностью беспомощны, если ничто динамически и интеллектуально не соединит релевантную информацию в контексте их задачи с информацией, к которой пользователи имеют доступ.

Если специалист в области информационных технологий не видит связи с релевантной частью информации на своем портале, или друг или коллега не посылают ему по электронной почте такую «связь», то информация удаляется; информация не подсоединяется и не приспособливается к пользовательскому контексту или к контексту, в котором она отображается. Аналогично, не достаточно только уведомить пользователя, что доступны новые данные для всего портала, и выгрузить их на локальный накопитель на жестком диске. Они не имеют настраиваемого представления с контекстно-зависимыми предупреждающими уведомлениями.

Семантическая сеть Web имеет те же ограничения, что и современная сеть Web, что касается контекстной зависимости. В семантической сети Web пользователи также находятся во власти создателей информации. Семантическая сеть сама будет иметь своего создателя, но такое создание будет включать в себя семантику. В результате пользователи по-прежнему в значительной степени предоставлены себе в определении местоположения и оценки релевантности доступной информации. Семантическая сеть Web, как автономный объект, не будет способна осуществлять такие динамические соединения с другими источниками информации.

Временная зависимость

Современная сеть Web не имеет временной зависимости. Платформа сети (например, браузер) является «неинтеллектуальной» частью программного обеспечения, которая просто представляет информацию без учета временной зависимости информации. Пользователь может делать логический вывод о временной зависимости или действовать без нее. Это приводит к огромным потерям производительности, поскольку платформа сети Web не может осуществлять в реальном времени соединения, использующие временную зависимость. Хотя некоторые Web-сайты фокусируются на представлении информации с временной зависимостью, например, индексируя информацию после заранее определенной даты, непосредственно Web-браузер не имеет понятия о временной зависимости. Вместо этого, индивидуальные Web-сайты могут включать временную зависимость в информацию, которую они отображают в их собственном «островке». Другими словами, на Web-связи отсутствует «ось времени».

Семантическая сеть Web подобно современной сети Web также не учитывает временную зависимость. Семантическая сеть Web может иметь семантические связи, которые не используют время. Это в значительной степени обусловлено тем, что семантическая сеть Web неявно не имеет понятия о программных услугах сети Web, которые рассматривают контекст и временную зависимость.

Автоматическая и интеллектуальная обнаружимость

Современная сеть Web не имеет автоматической и интеллектуальной обнаружимости для вновь создаваемой информации. В настоящее время нет способа узнать, какие Web-сайты начаты заново сегодня или вчера. Если только пользователь не уведомлен или случайно не обнаружит новый сайт при осуществлении поиска, то он может не иметь никакой подсказки относительно наличия новых Web-сайтов или страниц. Та же проблема существует на предприятиях. В интранете специалисты в области информационных технологий не имеют способа узнать, когда появятся новые сайты, если только не будут проинформированы через внешние средства. Сама платформа сети Web не имеет понятия об уведомлениях или обнаружении. Кроме того, нет контекстно-зависимого обнаружения для определения новых сайтов или страниц в контексте задачи пользователя или текущего информационного пространства.

Семантическая сеть Web, подобно современной сети Web, не рассматривает задачу отсутствия автоматической обнаружимости. Семантические Web-сайты имеют те же недостатки: пользователи должны выяснять о существовании новых информационных источников либо из внешних источников, либо посредством личного обнаружения при осуществлении поиска.

Динамическое связывание

Современная сеть Web использует для своей информационной модели «строгую» сеть или "структуру данных" в виде графа. Каждая Web-страница представляет узел в сети, и каждая страница может

содержать связи с другими узлами в сети. Каждая связь создается вручную на каждой странице. При этом есть несколько недостатков. Во-первых, это означает, что сеть должна поддерживаться, чтобы связь непрерывно имела значение. Если Web-страницы не обновлены или если Web-страница или авторы сайта не имеют порядка обслуживания для добавления связей к своим страницам на основе релевантности, сеть теряет значение. Современная сеть Web, по существу, склонна к наличию «тупиковых» связей, старых связей и т.д. Другой недостаток «строгой» сети или графовой информационной модели состоит в том, что потребитель информации находится скорее во власти представления Web-страницы или сайта, чем управляет им. Другими словами, если Web-страница или сайт не содержат какие-либо связи, пользователь не имеет ресурсов, чтобы найти релевантную информацию. Поисковые серверы являются слабой помощью, поскольку они просто возвращают страницы или узлы в сеть. Сама сеть не имеет какой-либо независимой или динамической возможности связывания. Таким образом, поисковый сервер может просто возвращать связи к Web-страницам, которые сами не имеют связей или имеют тупиковые, утратившие силу или несоответствующие связи. Как только пользователи получают результаты поиска, они предоставлены сами себе и полностью зависят от того, осуществил ли автор возвращенных страниц вставку в страницу релевантных, с временной зависимостью связей.

Семантическая сеть Web имеет тот же недостаток, что и современная сеть Web, потому что семантическая сеть Web является просто современной сетью Web «плюс семантика». Даже при том, что пользователи будут способны осуществлять навигацию по сети семантически (которую в настоящее время они не могут делать в сети), они будут по-прежнему зависеть от того, как была создана информация. Другими словами, семантическая сеть Web также зависит от порядка обслуживания создателей и, следовательно, имеет те же вышеупомянутые недостатки современной сети Web. Если семантическая сеть Web включает в себя страницы с онтологией и метаданными, но такие страницы не являются хорошо поддерживаемыми или не включают в себя связи с другими релевантными источниками, пользователь по-прежнему будет не способен получить имеющиеся связи и другую информацию. Семантическая сеть Web в ее нынешнем состоянии не будет интеллектуальной, динамической, самосоздаваемой, «самовосстанавливающейся» сетью.

Управляемые пользователем навигация и просмотр ресурсов

В современной сети Web пользователь не осуществляет управления навыками (опытом) навигации и просмотра ресурсов, а полностью зависит от того, как в Web-странице заданы связи (если они есть). Как показано со ссылкой на предшествующий уровень техники на фиг. 3, современная сеть Web состоит из "неинтеллектуальных связей" или статически создаваемых родовых связей, которые полностью зависят от непрерывного поддержания, чтобы осуществлять навигацию.

Семантическая сеть Web имеет подобный недостаток, как и современная сеть, в которой нет просмотра ресурсов, управляемого пользователем. Вместо этого, как показано со ссылкой на предшествующий уровень техники на фиг. 4, семантическая сеть Web состоит из "неинтеллектуальных связей", дополнительно включающих в себя семантическую информацию и метаданные. Однако связи семантической сети Web остаются зависящими от непрерывного поддержания для обеспечения возможности навигации.

Участие в сети «не HTML»-документов и локальных документов

Другим недостатком современной сети Web является требование, чтобы только документы, которые создают в формате HTML, могли участвовать в Web-сети, в дополнение к факту, что такие документы должны содержать связи. Следствием является то, что другие информационные объекты, подобные «не-HTML»-документам (например, PDF, Microsoft Word, PowerPoint, документы Excel и т.д.), особенно те, которые находятся на пользовательских накопителях на жестких дисках, исключены из преимуществ связывания с другими объектами в сети. Это является очень ограничивающим, особенно потому, что может существовать семантическая релевантность между информационными объектами, которые не являются HTML-документами и которые не содержат связей.

Кроме того, поисковые серверы не возвращают результаты для полной предметной области информации, так как огромное количество содержимого, доступного в сети, является недоступным для стандартных средств поиска («червяков») в сети. Это включает в себя, например, содержимое, сохраняемое в базах данных, неиндексированных файловых хранилищах, сайтах подписки, локальных машинах и устройствах, форматах файлов владельцев (таких как документы Microsoft Office и электронной почты), и нетекстовых мультимедийных файлах. Они формируют огромную совокупность недоступной предметной составляющей в Интернет, упоминаемую как "невидимый интранет" внутри корпораций. Современные серверы сети Web не обеспечивают сеть инструментальными средствами web-поиска для решения этой проблемы.

Семантическая сеть Web также имеет это ограничение. Она не учитывает миллионы «не-HTML»-документов, которые уже выпущены, особенно те, которые находятся на пользовательских накопителях на жестких дисках. Следствием является то, что документы, которые не имеют эквивалентов метаданных RDF или программ-посредников, не могут быть динамически связаны с сетью.

Гибкое представление, которое интеллектуально передает семантику отображаемой информации

Современная сеть Web не позволяет пользователям настраивать или формировать "поверхность" Web-сайта или страницы. Это объясняется тем, что современные Web-серверы возвращают информацию, которая уже отформатирована для представления браузером. Конечный пользователь не имеет гибкости в выборе лучших средств отображения информации на основании различных критериев (например, типа информации, доступного пространства на видеодиске и т.д.).

Семантическая сеть не решает вопрос гибкого представления. Хотя сайт семантической сети Web концептуально использует RDF и онтологии, он по-прежнему посылает HTML-документы браузеру. По существу, семантическая сеть Web не предусматривает для конкретного пользователя полномочия для представления. Как таковой сайт семантической сети Web, рассматриваемый платформой современной сети Web, по-прежнему не будет давать пользователю возможность гибкого представления. Более того, несмотря на продвижение развития в направлении использования XML, только новая платформа может обуславливать, что данные будут отделены от представления, и определять рекомендации, чтобы делать данные программируемыми. Разработчики, создающие содержимое для семантической сети Web, либо возвращают XML и полностью избегают вопросов представления, либо сосредотачивают свои усилия на едином стиле представления (вертикальный промышленный сценарий) для визуального воспроизведения. Никакой подход не позволяет семантической сети Web достичь оптимальной степени распространения знаний.

Логика, логический вывод и умозаключение

Поскольку современная сеть Web не имеет семантики, метаданных, или представления знаний, компьютеры не могут обрабатывать Web-страницы, используя логику и логический вывод, чтобы вывести новые связи, выдавать уведомления, и т.д. Современная сеть Web была разработана и построена для использования человеком, а не для использования компьютером. Как таковая, современная сеть Web не может действовать на информационной «ткани» без обращения к хрупким, ненадежным способам типа «соскоба» экрана, чтобы попытаться извлекать метаданные и применять логику и логический вывод.

Хотя семантическая сеть Web концептуально использует метаданные и значение для обеспечения Web-страниц и сайтов кодируемой информацией, которая может быть обработана компьютерами, отсутствует реализация, которая успешно реализовала бы такую компьютерную обработку и которая обеспечила бы новые или усовершенствованные сценарии с пользой для потребителя или производителя информации.

Гибкий управляемый пользователем анализ информации

Современная сеть Web не имеет управляемого пользователем анализа информации. Современная сеть Web не позволяет пользователям отображать различные "виды" связей, используя различные фильтры и условия. Например, поисковые серверы сети Web не дают возможности пользователям проверять результаты поисков при различных сценариях. Пользователи не могут просматривать результаты, используя различные «точки опоры», такие как тип информации (например, документы, электронная почта и т.д.), контекст (например, "заголовки", "лучшие выборы" и т.д.), категория (например, "беспроводная", "способ" и т.д.) и т.д.

Обеспечивая большую степень гибкого анализа информации, семантическая сеть Web не описывает, как уровень представления может взаимодействовать непосредственно с сетью Web интерактивным образом для обеспечения гибкого анализа.

Гибкие семантические запросы

Современная сеть Web допускает запросы только на основе текста или запросы, которые привязаны к схеме конкретного сайта. Таким запросам не хватает гибкости. Современная сеть Web не дает возможности пользователю осуществлять запросы на языке, приближенном к естественному, или включать семантику и локальный контекст. Например, запрос такой, как "Найти мне все сообщения электронной почты, написанные моим начальником или кем-либо по исследовательской работе, и которые относятся к этому описанию на моем жестком диске" в современной сети Web не возможен.

Применяя метаданные и онтологии, концептуальная семантическая сеть Web дает возможность пользователю осуществлять более гибкие запросы, чем современная сеть Web. Например, пользователи смогут сделать запрос типа "Найти мне все сообщения электронной почты, написанные моим начальником или кем-либо по исследовательской работе". Однако пользователи не смогут включать в него локальный контекст. Кроме того, семантическая сеть Web не определяет простой способ, с помощью которого пользователи будут запрашивать сеть Web без использования естественного языка. Способ общения на естественном языке является вариантом выбора, но далек от того, чтобы быть надежным способом. Необходим пользовательский интерфейс запросов на приближенном к естественному языке, но еще не основывающийся на естественном языке. Семантическая сеть Web не решает эту проблему.

Поддержка считывания/записи

Современная сеть Web является сетью только для считывания. Например, если пользователи обнаруживают тупиковую связь (например, посредством ошибки "404"), они не могут "устанавливать" связь, указывая ее на обновленный адресат, который может быть известен пользователю. Это может являться

ограничением, особенно в тех вариантах, когда пользователи имеют существенные знания, которые могут быть предоставлены другим, и когда пользователям желательно ввести данные относительно того, как сеть следует представлять и развивать.

Хотя семантическая сеть Web концептуально допускает сценарии для считывания/записи, как предусмотрено независимыми участвующими приложениями, однако, в настоящее время нет реализации, которая обеспечивает такую возможность.

Аннотации

Современная сеть Web не имеет неявной поддержки для аннотаций. И хотя некоторые конкретные сайты поддерживают аннотации, они делают это очень ограниченным и автономным способом. Современная среда сети Web не рассматривает аннотации. Другими словами, для пользователей нет возможности аннотировать произвольную связь со своими комментариями или дополнительной информацией, к которой они имеют доступ. Это приводит к возможной потере информации.

Хотя семантическая сеть концептуально предусматривает, что аннотации будут встроены в систему из соображений защищенности, в настоящее время нет реализации, которая обеспечивает такую возможность.

«Сеть доверия»

В современной сети Web отсутствует «бесшовная» (непрерывная) интеграция аутентификации, управления доступом и авторизации в сети Web, т.е. того, что определяется «Сеть доверия». В сети доверия, например, пользователи имеют возможность устанавливать условия, устанавливать и обновлять связи к сети и иметь ограничения управления доступом, встроенные для таких операций. В современной сети Web, этот недостаток доверия означает также, что услуги сети Web остаются независимыми «островками», которые должны реализовывать авторизацию подписки пользователя на собственность, управление доступом или систему оплаты. Большие схемы для централизации такой информации на серверах третьих сторон сталкиваются с недоверием потребителя и продавца из-за опасений секретности. Чтобы получать доступ к мощной (по содержанию) информации, пользователи активнов должны входить в систему индивидуально и обеспечивать информацию идентичности на каждом сайте.

Хотя семантическая сеть Web концептуально предусматривает сеть доверия, в настоящее время отсутствует реализация, которая обеспечивает такую возможность.

Информационные пакеты (элементы сопряжения)

Ни современная сеть Web, ни семантическая сеть Web не дают пользователям возможность использовать зависимую семантическую информацию как целый модуль, посредством объединения характеристик для потенциально различающейся семантической информации, чтобы выработать перекрывающиеся результаты (например, подобно созданию заказной, личной газеты или канала телевидения).

Контекстные шаблоны

Ни современная сеть Web, ни семантическая сеть Web не дают пользователям возможности независимого создания и отображения на конкретные и известные семантические модели для доступа и поиска информации.

Ориентированное на пользователя агрегирование информации

Современная сеть Web не имеет поддержки для ориентированного на пользователя агрегирования информации. Пользователь может только осуществлять доступ к одному сайту или к одному поисковому серверу одновременно, в контексте одного сеанса просмотра ресурсов сети. Даже если есть контекстная или с временной зависимостью информация в других источниках информации, которая относится к просматриваемой пользователем информации, такие источники не могут быть представлены целостным образом в текущем контексте задачи пользователя.

Семантическая сеть Web также не обеспечивает ориентированного на пользователя агрегирования информации. Сама среда является расширением современной сети Web. Как таковые, пользователи будут по-прежнему осуществлять доступ к одному сайту или одному поисковому серверу в каждый данный момент времени и не будут иметь возможность агрегировать информацию из информационных хранилищ контекстным методом или с использованием временной зависимости.

Учитывая возрастающую потребность в получении "знаний на кончиках пальцев", а также недостатки современной сети Web и концептуальной семантической сети Web, многие из которых отмечены выше, существует необходимость в новой и всеобъемлющей системе и способе поиска, управления и доставки знаний.

Краткое описание сущности изобретения

Настоящее изобретение направлено в одном аспекте на интегрированную и «бесшовную» (непрерывную) реализацию базовой структуры и получаемой в результате среды для поиска, управления, доставки и представления знаний. Система включает в себя сервер, состоящий из нескольких компонентов, которые функционируют совместно для обеспечения услуг контекстного и с учетом временных зависимостей поиска семантической информации для клиентов, оперирующих с платформой представления через коммуникационную среду. Сервер включает в себя первый серверный компонент, который обеспечивает добавление и поддержание проблемно-зависимой семантической информации или сведений. Первый серверный компонент предпочтительно включает в себя структуру или методологию, направленную

на обеспечение следующего: семантической сети, блока сбора семантических данных, блока проверки согласованности семантической сети, машины (логического) вывода, процессора семантических запросов, синтаксического анализатора естественного языка, агента знаний электронной почты и администратора знаний предметной области. Сервер включает в себя второй серверный компонент, который ведет (содержит) проблемно-зависимую информацию, используемую для классификации и категоризации семантической информации. Первый и второй серверные компоненты действуют совместно и могут быть физически объединены или разделены.

Внутри системы все объекты или события в заданной иерархии являются активными агентами, семантически связанными между собой и представляющими запросы (состоящие из базовых кодов операций), которые возвращают объекты данных для представления клиенту согласно предварительно определенной и настраиваемой теме, или "поверхности" (оболочке). Эта система обеспечивает различные средства для клиента для настройки и "сопряжения" агентов и соответствующих базовых запросов для оптимизации представления результирующей информации.

Сквозная системная архитектура настоящего изобретения обеспечивает средства доступа множества клиентов для связи между источниками информации разнообразных знаний через независимую платформу семантической сети Web или через традиционный портал сети Web (например, современный браузер доступа к сети Web), как это усовершенствовано посредством настоящего изобретения, обеспечивающего дополнительные уровни SDK (набора инструментальных средств разработки программного обеспечения), которые допускают программную интеграцию со специализированным клиентом.

Методология настоящего изобретения направлена частично на операционные аспекты всей системы, включая поиск, управление, доставку и представление знаний. Это предпочтительно включает в себя обеспечение защиты информации из источников информации, семантическое связывание информации из источников информации, поддержание семантических свойств внутренней информационной части (тела) семантически связанной информации, доставку требуемой семантической информации на основании пользовательских запросов и представления семантической информации в соответствии с настраиваемыми пользовательскими предпочтениями. Альтернативные варианты воплощения методологии настоящего изобретения направлены на функционирование агентов, представляющих запросы, которые используются серверными и клиентскими приложениями для обеспечения эффективных, основанных на логическом выводе запросов, формирующих семантически релевантную информацию.

Краткое описание чертежей

Предпочтительные и альтернативные варианты воплощения настоящего изобретения подробно описаны ниже со ссылкой на последующие чертежи, на которых

- фиг. 1 - таблица, показывающая технологические уровни современной сети Web;
- фиг. 2 - таблица, показывающая технологические уровни средств концептуальной семантической сети Web;
- фиг. 3 - схема, показывающая навигацию пользователя по связям в современной сети Web;
- фиг. 4 - схема, показывающая навигацию пользователя по связям в концептуальной семантической сети Web;
- фиг. 5 - вид экрана, показывающий пример панели результатов информационного агента в соответствии с настоящим изобретением;
- фиг. 6 - стеки технологической платформы современной сети Web и информационной «нервной» системы настоящего изобретения;
- фиг. 7 - схема, показывающая общее представление системы настоящего изобретения;
- фиг. 8 - схема, показывающая сквозную системную архитектуру для информационной «нервной» системы настоящего изобретения;
- фиг. 9 - схема, показывающая системную архитектуру для сервера интеграции знаний (СИЗ) информационной «нервной» системы настоящего изобретения;
- фиг. 10 - сравнение между высокоуровневыми описательными уровнями платформы современной сети Web и эквивалентов (если применимы) в информационной «нервной» системе настоящего изобретения;
- фиг. 11 - предпочтительное воплощение информационной «нервной» системы и неоднородный, межплатформенный контекст для настоящего изобретения;
- фиг. 12-14 - примерные виды экранов для разных аспектов пользовательского интерфейса мастера элемента сопряжения согласно предпочтительному воплощению настоящего изобретения;
- фиг. 15 - пример панели пользовательского интерфейса агента новостей;
- фиг. 16 - пример предпочтительного воплощения, показывающего диалог открытия агентов согласно настоящему изобретению;
- фиг. 17-19 - представление в виде дерева примера экземпляра семантической среды, использующей диалог открытия агента;
- фиг. 20 - схема агента для предпочтительного воплощения настоящего изобретения;
- фиг. 21 - AgentTypeIDs (идентификаторы типа агента) для предпочтительного воплощения настоящего изобретения;

- фиг. 22 - AgentQueryTypeIDs (идентификаторы типа запроса агента) предпочтительного воплощения настоящего изобретения;
- фиг. 23 - примеры семантических запросов, которые соответствуют именам агентов, показывающим, как серверные агенты предпочтительно конфигурированы в СИЗ настоящего изобретения;
- фиг. 24 - схема, показывающая общее представление для СИЗ согласно настоящему изобретению;
- фиг. 25 - схема, показывающая пример семантической сети, ориентированную согласно условиям предприятия в соответствии с настоящим изобретением;
- фиг. 26 - таблица, показывающая предпочтительную схему типа объекта в соответствии с настоящим изобретением;
- фиг. 27 - таблица семантических связей для настоящего изобретения;
- фиг. 28 - таблица, показывающая ИД предикатных типов предпочтительного воплощения настоящего изобретения;
- фиг. 29 - таблица, показывающая предпочтительную схему объекта «пользователь», осуществленную в соответствии с настоящим изобретением;
- фиг. 30 - таблица, показывающая MailingAddressTypeIDs (идентификаторы типа почтового адреса), предпочтительно связанные со схемой объекта «пользователь» (персона);
- фиг. 31 - таблица предпочтительной схемы объекта «категория», осуществленной в соответствии с настоящим изобретением;
- фиг. 32 - таблица предпочтительной схемы объекта «документ», осуществленной в соответствии с настоящим изобретением;
- фиг. 33 - ИД типов печатных средств информации для предпочтительного воплощения;
- фиг. 34 - предпочтительный FORMATTYPEID (идентификатор типа формата);
- фиг. 35 - предпочтительная схема объекта «перечень сообщений электронной почты», осуществленная в соответствии с настоящим изобретением;
- фиг. 36 и 37 - примерные таблицы, показывающие, соответственно, схемы объектов «перечень распространения электронной почты» и «общедоступная папка электронной почты» предпочтительного воплощения настоящего изобретения;
- фиг. 38 - предпочтительный PublicFolderTypeID (идентификатор типа общедоступной папки) для настоящего изобретения;
- фиг. 39 - предпочтительная схема объекта «перечень сообщений» для схемы объекта «событие», осуществленная в соответствии с настоящим изобретением;
- фиг. 40 - типы событий для предпочтительного воплощения настоящего изобретения;
- фиг. 41 - предпочтительная схема объекта «перечень сообщений» для схемы объекта «средства информации», осуществленная в соответствии с настоящим изобретением;
- фиг. 42 - типы средств информации предпочтительного воплощения настоящего изобретения;
- фиг. 43-45 - дополнительные примеры, показывающие, как определяются категории и используются объекты в предпочтительном воплощении настоящего изобретения;
- фиг. 46 - граф объекта, показывающий отображение строки метаданных XML для электронной почты на семантическую сеть согласно настоящему изобретению;
- фиг. 47-53 - примеры видов экранов, показывающие особенности управления агентами посредством СИЗ;
- фиг. 54 - пример пользовательского интерфейса, иллюстрирующего информационный объект, отображаемый в панели результатов информационного агента;
- фиг. 55 - пример всплывающего контура, ассоциируемого с внутренней семантической связью, показывающий пример электронной почты согласно настоящему изобретению;
- фиг. 56 - пример всплывающего контура, ассоциируемого с пользовательским интерфейсом «команда действий» (в форме глагольного императива) согласно настоящему изобретению;
- фиг. 57 - пример всплывающего контура, ассоциируемого с пользовательским интерфейсом режим «глубокой информации» согласно настоящему изобретению;
- фиг. 58 и 59 - иллюстрации, показывающие примерную семантическую среду согласно настоящему изобретению;
- фиг. 60-68 - примерные виды экранов для информационного агента согласно предпочтительному воплощению настоящего изобретения;
- фиг. 69-71 - пример всплывающего меню, ассоциируемого с признаком «интеллектуальная лупа» информационного агента согласно настоящему изобретению;
- фиг. 72 - пример варианта всплывающего меню по фиг. 71, показывающего меру связанности двух объектов;
- фиг. 73-75 - примеры таблиц, иллюстрирующих поведения и реляционные предикаты типов объектов содержания при использовании «интеллектуальной лупы»;
- фиг. 76 - пример пользовательского интерфейса, иллюстрирующий соответствующие настоящему изобретению семантические результаты для элемента управления воспроизведением/предварительным просмотром;

фиг. 77 - пример пользовательского интерфейса, показывающего семантические результаты для элемента сопряжения;

фиг. 78 и 79 - примерные отображения функциональных возможностей для настоящего изобретения;

фиг. 80 - пользовательский интерфейс, показывающий результаты агента и соответствующие палитры контекста согласно настоящему изобретению;

фиг. 81 - пример панели результатов всплывающих контекстных интеллектуальных рекомендаций согласно настоящему изобретению;

фиг. 82 - таблица, показывающая технологические уровни информационной «нервной» системы для настоящего изобретения;

фиг. 83 - динамическое связывание и управляемая пользователем навигация и просмотр ресурсов сети согласно предпочтительному воплощению настоящего изобретения.

Документы, включенные посредством ссылки

Приложение к настоящему описанию, которое содержит ссылки на него, включено в настоящее описание посредством ссылки. Данное приложение включает в себя пример кода, иллюстрирующего предпочтительный вариант осуществления настоящего изобретения.

Содержание детального описания изобретения

- A. Определения
- B. Обзор
 - 1. Контекст изобретения
 - 2. Значение задач
 - 3. Современная сеть Web в сравнении с информационной «нервной» системой согласно настоящему изобретению
- C. Архитектура системы и технологические факторы
 - 1. Обзор системы
 - 2. Архитектура системы
 - 3. Технологические наборы
 - 4. Гетерогенность системы
 - 5. Защищенность
 - 6. Факторы эффективности
- D. Компоненты и функционирование системы
 - 1. Агентства и агенты
 - a. Агентства
 - b. Агенты
 - 2. Сервер интеграции знаний
 - a. Семантическая сеть
 - b. Блок сбора семантических данных
 - c. Блок проверки согласованности семантической сети
 - d. Машина логического вывода
 - e. Процессор семантических запросов
 - f. Синтаксический анализатор естественного языка
 - g. Агент знаний электронной почты
 - h. Администратор знаний предметной области
 - i. Другие компоненты
 - 3. Сервер базы знаний
 - 4. Информационный агент (платформа браузера семантики)
 - a. Обзор
 - b. Конфигурация клиента
 - c. Спецификация основы клиента
 - d. Основа клиента
 - e. Документ семантического запроса
 - f. Семантическая среда
 - g. Администратор семантической среды
 - h. Браузер среды (браузер семантики или информационный агент™)
 - i. Дополнительные особенности приложений
 - 5. Обеспечение контекста в настоящем изобретении
 - a. Шаблоны контекста
 - b. Поверхности контекста
 - c. Шаблоны поверхности
 - d. Предикаты, устанавливаемые по умолчанию
 - e. Предикаты контекста
 - f. Признаки контекста

- g. Палитры контекста
- h. Внутренние уведомления
- i. Интеллектуальные рекомендации
- 6. Преимущества свойств настоящего изобретения
- Е. Сценарии
 1. Примеры использования семантических запросов согласно настоящему изобретению
 2. Проблемы коммерции
 3. Ситуации

Детальное описание изобретения

А. Определения.

ActionScript (скрипт действий) - язык сценариев Macromedia Flash (групповая запись/считывание макросреды). Этот двунаправленный информационный обмен способствует созданию пользователями интерактивных фильмов. См. http://www.macromedia.co/support/flach/action_scripts/actionsript_tutorial/.

Агентство - имеющий имя экземпляр сервера интеграции знания (СИЗ), который является семантическим эквивалентом Web-сайта.

Каталог агентства - каталог, который хранит информацию метаданных для агентств и позволяет клиентам осуществлять добавление, удаление, поиск, просмотр ресурсов агентств, сохраненных в нем. Агентства могут публиковаться в каталогах подобных LDAP (Облегченный протокол службы каталогов) или активном каталоге Microsoft. Агентства могут также публиковаться в частных каталогах, построенных специально для агентств.

Агент - семантический фильтр-запрос, который возвращает XML-информацию для конкретного типа семантического объекта (например, документов, электронной почты, людей и т.д.), контекста (заголовки, диалоги и т.д.) или элемента сопряжения.

Blender™ или Compound Agent™ (агент элемента сопряжения или составной агент) - зарегистрированное имя для агента, который содержит других агентов и позволяет пользователю (в случае элементов сопряжения на клиентской стороне) или администратору агентства (в случае элементов сопряжения на серверной стороне) создавать запросы, которые генерируют результаты, которые являются объединением или пересечением результатов содержащихся в них агентов. В случае элементов сопряжения на клиентской стороне результаты могут генерироваться с использованием различных видов (показывающих каждого агента в элементе сопряжения в отдельном кадре, показывающих все объекты конкретного типа объекта среди содержащихся агентов и т.д.).

Breaking News Agent™ (агент новостей, приводящих к прерыванию) - зарегистрированное имя для интеллектуального агента, который пользователи специально маркируют как указывающие на критичность по времени. Пользователи могут маркировать любого интеллектуального агента как агента новостей, приводящих к прерыванию. Этот признак затем сохраняется в семантической среде пользователя. Агент новостей, приводящих к прерыванию предпочтительно показывает уведомление, если имеются новости, приводящие к прерыванию, относящиеся к какой-либо отображаемой информации.

Default Agent™ (устанавливаемый по умолчанию Агент) - зарегистрированное имя для стандартизованного, не модифицируемого пользователем агента, представляемого пользователю.

Domain Agent™ (агент предметной области) - зарегистрированное имя для агента, который принадлежит семантической предметной области. Он инициализируется запросом агента, который включает в себя ссылку на таблицу «категории».

Dumb Agent™ («немой» (неинтеллектуальный) агент) - зарегистрированное имя для агента, который не имеет агентства и который ссылается на локальную информацию (на локальном жестком диске), на совместно используемую сеть или на Web-связь (линк) или URL (унифицированный указатель информационного ресурса). Неинтеллектуальные агенты используются, по существу, для загрузки единиц информации (например, документов) из неинтеллектуальной ограниченной среды («песочницы») в интеллектуальную ограниченную среду (в информационную «нервную» систему через информационного агента (семантический браузер)).

Email Agent™ (или Email Knowledge Agent™) (агент электронной почты или агент знаний электронной почты) - зарегистрированные имена для общественного агента, используемого для публикации или аннотирования информации и совместного использования знаний в агентстве.

Favorite Agent™ (любимый агент) - зарегистрированное имя для агентов, которых пользователи указывают как любимые и как те, к которым наиболее часто обращаются.

Public Agent™ (общественный агент) - зарегистрированное имя для агентов, которые создаются и управляются системным администратором.

Private или Local Agents™ (частные или локальные агенты) - зарегистрированные имена для агентов, которые создаются и управляются пользователями.

Search Agent™ (агент поиска) - зарегистрированное имя для интеллектуального агента, который создается путем поиска в семантической среде с ключевыми словами или поиска существующего интеллектуального агента для вызова дополнительного фильтра текстового запроса в интеллектуальном аген-

те.

Simple или Standard Agent™ (простой или стандартный агент) - зарегистрированное имя автономных агентов, которые инкапсулируют структурированные несемантические запросы (например, от локальной файловой системы или источника данных).

Smart Agent™ (интеллектуальный агент) - зарегистрированное имя автономного агента, который инкапсулирует структурированные семантические запросы и ссылается на агентство через свой XML-Web-сервис.

Special Agent™ (специальный агент) - зарегистрированное имя для интеллектуального агента, который создается на основе шаблона контекста.

Обнаружение агента - свойство информационной среды настоящего изобретения, которое позволяет пользователям просто и автоматически обнаруживать новых агентов серверной стороны или агентов клиентской стороны, созданных другими (друзьями или коллегами). См. также «обнаружимость».

Аннотации - примечания, комментарии или пояснения, которые используются для добавления персонального контекста к информационному объекту. В предпочтительном варианте осуществления аннотации представляют собой сообщения электронной почты, которые связаны с объектом, который они определяют, и которые могут иметь приложения (подобно обычным сообщениям электронной почты). Кроме того, аннотации являются информационными объектами первого класса в системе и как таковые могут аннотироваться сами, приводя в результате к дереву аннотаций с исходным объектом в качестве корня.

Интерфейс прикладного программирования (API) - определяет, каким образом программисты используют конкретные компьютерные функции. API существуют для оконных систем, файловых систем, систем баз данных, сетевых систем и других систем.

Протокол доступа к календарю (CAP) - протокол Интернет, позволяющий пользователям получать цифровой доступ к памяти календаря на основе iCalendar-стандарта.

Compound Agent Manager™ (администратор составного агента) - зарегистрированное имя для компонента агента, который программным образом позволяет пользователю создавать и удалять составных агентов и управлять ими путем добавления и удаления агентов.

Контекст – информация, окружающая конкретный элемент (объект, предмет), которая придает значение и иным образом помогает потребителю информации интерпретировать этот объект, а также находить другую релевантную информацию, относящуюся к этому объекту.

Панель результатов контекста - панель результатов, которая отображает результаты запросов на основе контекста (контекстно-основанных запросов). Сюда включаются палитры контекста, интеллектуальные лупы, глубокая информация и т.д. См. «панель результатов».

Чувствительность к контексту - свойство информационной среды, которое позволяет ей интеллектуально и динамически воспринимать контекст всего, что представляет информация, и представлять дополнительную релевантную информацию для данного контекста. Контекстно-чувствительная система или среда понимает семантику информации, которую она представляет и обеспечивает соответствующее поведение (проактивное и реактивное, на основе действий пользователя), чтобы представить информацию в ее надлежащем контексте (как внутренним, так и относительным (реляционным) образом).

Context Template™ (шаблон контекста) - зарегистрированное имя для управляемых сценарием шаблонов запроса информации, которые отображаются на конкретные и знакомые семантические модели для доступа к информации и ее извлечения. Например, шаблон «заголовки» в предпочтительном варианте имеет параметры, которые согласованы с доставкой «заголовков» (где новизна и вероятность интереса высокого уровня являются основными параметрами для извлечения). Шаблон «наступающие события» имеет параметры, которые согласованы с доставкой «наступающих событий» и т.д. По существу, шаблоны контекста могут быть представлены по аналогии как «каналы» извлечения персональной цифровой семантической информации, которые доставляют информацию пользователю путем использования хорошо известного семантического шаблона.

Deep Information™ (глубокая информация) - зарегистрированное имя для признака настоящего изобретения, который позволяет информационному агенту отображать внутреннюю контекстную информацию, относящуюся к информационному объекту. Контекстная информация включает информацию, отыскиваемую из семантической сети агентства, откуда исходит объект.

Обнаружимость - способность информационной среды настоящего изобретения интеллектуальным и проактивным (упреждающим) образом делать информацию известной или видимой пользователю, не требуя от пользователя в явном виде отыскивать информацию.

Domain Agent Wizard™ (мастер агента предметной области) - зарегистрированное имя для системного компонента и его пользовательского интерфейса, позволяющих администратору агентства создавать и управлять агентами предметной области.

DOTNET (.NET) - Mirisoft®.Net является сетью технологий программного обеспечения Microsoft для соединения информации, людей, систем и устройств. Она обеспечивает интеграцию программного обеспечения за счет использования XML-Web-сервисов: малых, дискретных компонентных приложений,

которые соединяются друг с другом, а также с другими более крупными приложениями, через Интернет. Net-соединенное программное обеспечение облегчает создание и интеграцию XML-Web-сервисов. См. <http://www.microsoft.com/net/defined/default.asp>.

Dynamic Linking™ (динамическое связывание) - зарегистрированное имя для свойства информационной нервной системы согласно настоящему изобретению, позволяющего пользователям связывать информацию динамически, семантически и со скоростью мысли, даже если эти информационные объекты не содержат связей друг с другом. За счет использования интеллектуальных объектов, которые обладают внутренне присущим им поведением, и использования рекурсивной информации, содержащейся в XML-Web-сервисе информационного агентства, каждый узел в семантической сети является намного более интеллектуальным, чем регулярная связь или узел современной Web-сети или концептуальной семантической Web-сети. Иными словами, каждый узел в интеллектуальной виртуальной сети или Web-сети согласно настоящему изобретению может связываться с другими узлами независимо от авторских разработок. Каждый узел имеет поведение, которое может динамически связываться с агентством и интеллектуальными агентами через операции графического интерфейса «перетащить и оставить» и интеллектуального копирования и вставки, создавать связи с агентствами в семантической среде, отвечать на запросы от интеллектуальных агентов для создания новых связей, включать внутренние уведомления, которые будут динамически создавать связи с контекстной и чувствительной к времени информации в своем агентстве, включая представление подсказок для новостей, вызывающих прерывание (причем узел может автоматически связываться с агентами новостей, вызывающих прерывание, в пространстве имен), образовывать базу для глубокой информации, которая может позволить пользователю найти новые связи и т.д. Пользователь настоящего изобретения поэтому не зависит от автора метаданных. Как только пользователь достигает узла в сети, пользователь имеет множество семантических средств навигации динамическим и автоматическим образом с использованием контекста, времени, связанности с интеллектуальными агентствами и агентами и т.д.

XML-объект электронной почты - информационный объект с типом объекта информации «Электронная почта». XML-объект имеет SRML-схему «Электронной почты» (которая использует XML).

Браузер среды - см. информационный агент.

Favorite Agents Manager™ (администратор любимых агентов) - зарегистрированное имя для системного компонента и элемента пользовательского интерфейса, который позволяет администратору агентства управлять любимыми агентами серверной стороны.

Flash MX - Macromedia Flash MX (групповая запись/считывание макросреды) является средой проектирования и разработки текста, графики и анимации для создания широкого диапазона содержимого высокой степени воздействия и обогащенных приложений для Интернет.

См. http://www.macromedia.com/software/flash/productinfo/product_overview/.

Global Agency Directory™ (глобальный каталог агентств) - зарегистрированное имя для экземпляра каталога агентств, который исполняется в Интернет (или иной глобальной сети). Глобальный каталог агентств позволяет пользователям осуществлять поиск и просмотр ресурсов агентств, базирующихся на Интернет, с использованием их информационного агента (непосредственно в их семантической среде). См. также каталог агентств.

HTTP - протокол передачи гипертекстовых файлов является протоколом уровня приложений для распределенных, взаимодействующих гипермедийных информационных систем. Это обобщенный, не изменяющий своего состояния в процессе исполнения протокол, который может быть использован для многих задач, помимо его использования для гипертекста, например, серверов имен и систем управления распределенными объектами, за счет расширения его методов запроса, кодов ошибок и заголовков. Особенностью HTTP является типизация и согласование представления данных, позволяющие системам формироваться независимо от переносимых данных. См. <http://www.w3org/Protocols> и <http://www.w3org/Protocols.Specs.html>.

Inference Engine™ (машина логического вывода) - зарегистрированное имя для методологии настоящего изобретения, которая соблюдает шаблоны и данные, чтобы прийти к релевантным и логически правильным выводам путем умозаключений. Предпочтительно использует правила логического вывода (предварительно определенный набор эвристических правил) для добавления семантических связей к семантической сети согласно настоящему изобретению.

Информация - количественная или качественная мера релевантности и уровня интеллекта содержимого или данных, которая передает знание.

Information Agent™ (информационный агент) - зарегистрированное имя для семантического клиента или браузера согласно настоящему изобретению, который обеспечивает контекстную и чувствительную к времени доставку и представление анализируемой информации (или знания) из множества источников, типов информации и шаблонов и который обеспечивает возможность динамического связывания информации из различных хранилищ.

Information Nervous System™ (информационная нервная система) - зарегистрированное имя для динамической, самосоздаваемой, контекстной и чувствительной к времени информационной системы со-

гласно настоящему изобретению, которая позволяет пользователям интеллектуальным образом и динамически связывать информацию со скоростью мысли, в контексте и с временной зависимостью, чтобы максимизировать сбор и использование знания для решаемой задачи.

Information Object™ (информационный объект, или элемент, или пакет) - зарегистрированное имя для единицы информации конкретного типа, которая переносит знание в заданном контексте.

Information Object Pivot™ (опорный пункт информационного объекта) - зарегистрированное имя информационного объекта, который пользователи используют в качестве навигационного опорного пункта для нахождения другой релевантной информации в том же самом контексте.

Тип информационного объекта - см. тип объекта.

Интеллектуальный объект - агенты программного обеспечения, которые действуют от имени пользователя для нахождения и фильтрации информации, согласования услуг, простой автоматизации комплексных задач, сотрудничества с другими агентами программного обеспечения для решения комплексных проблем. По определению, интеллектуальные агенты должны быть автономными или, иными словами, свободно способными к исполнению без вмешательства пользователя. Кроме того, интеллектуальные агенты должны быть способны осуществлять информационный обмен с другим программным обеспечением или человеческими агентами и должны иметь возможность воспринимать и контролировать среду, в которой они находятся.

См. http://www.findarticles.com/cf_dls/mOFWE/7_4/64694222/p1/article.jhtml.

Internet Calendaring and Scheduling (iCalendar) (Регистрация и планирование в Интернет) - протокол, который позволяет развертывание взаимодействующих услуг регистрации и планирования для Интернет. Протокол обеспечивает определение общего формата для открытого обмена информацией регистрации и планирования в Интернет.

Internet Message Access Protocol (IMAP) (Протокол доступа к сообщениям в сети Интернет) - коммуникационный механизм для почтовых клиентов для взаимодействия с почтовыми серверами и манипулирования почтовыми ящиками на них. Возможно, наиболее популярным протоколом доступа к почтовым сообщениям в настоящее время является протокол POP (почтовый протокол), который также обслуживает потребности удаленного доступа к почтовым сообщениям. Протокол IMAP предоставляет супернабор характеристик протокола POP, который позволяет намного более сложные взаимодействия и обеспечивает намного более эффективный доступ, чем модель POP.

См. <http://222-smi.Stanford.edu/projects/imap/m1/imap.html>.

Intrinsic Semantic Link™ (внутренняя семантическая связь) - зарегистрированное имя для семантических связей, которые являются внутренними для схемы конкретного информационного объекта. Например, информационный объект электронной почты имеет внутренние связи «от», «к», «с», «вс», «приложения», которые являются присущими самому объекту и определены в схеме для типа информационного объекта «электронная почта».

Островок - хранилище информации, которое изолировано от других хранилищ, которые могут содержать релевантную, семантически связанную, контекстную и чувствительную к времени информацию, но которые отсоединены от других контекстов, в которых такая информация может быть релевантной.

J2EE - платформа Java™ 2, издание корпоративного уровня (издание пакета для предприятий) (J2EE), используемое для разработки многоуровневых (многозвенных) корпоративных приложений. J2EE основывает корпоративные приложения на стандартизованных, модульных компонентах путем обеспечения набора сервисов для этих компонентов и путем автоматической обработки многих деталей линии поведения приложения. См. <http://Java.sun.com.j2ee/overview.html>.

Знание - информация, представленная в контекстной и зависимой от времени форме, которая позволяет потребителю информации обучаться, исходя из этой информации, и применять эту информацию, чтобы принимать более обоснованные и своевременные решения для релевантных задач.

Knowledge Agent™ (агент знания) - см. информационный агент.

Knowledge Base Server™ (сервер базы знаний) - зарегистрированное имя для сервера, на котором находятся знания для сервера интеграции знаний (СИЗ).

Knowledge Domain Manager™ (администратор предметной области знаний) - зарегистрированное имя для компонента сервера интеграции знаний, который ответственен за пополнение и поддержание специфической для предметной области информации в семантической сети.

Knowledge Integration Server™ (сервер интеграции знаний) - зарегистрированное имя для сервера, который семантически интегрирует данные из множества разнообразных источников в семантическую сеть, который может также содержать агентов серверной стороны, которые обеспечивают доступ к сети, и который содержит XML-Web-сервисы, обеспечивающие контекстный и зависящий от времени доступ к знаниям на сервере.

Knowledge Web™ - см. информационная нервная система.

Liberty Alliance (свободный альянс) - мировоззренческая концепция свободного альянса состоит в обеспечении возможности связанному сетью сообществу более просто проводить транзакции при обеспечении защиты от пиратства и защищенности критичной информации идентификации. Для реализации

своей концепции свободный альянс стремится устанавливать открытый стандарт для интегрированного сетевого сообщества через открытые технические спецификации.

См. <http://www.projectliberty.org/index.html>.

Lightweight Directory Access Protocol (LDAP) (облегченный протокол службы каталогов) - технология для доступа к общей информации каталога. LDAP заключается и реализуется в наиболее ориентированном на сеть промежуточном программном обеспечении (связующее программное обеспечение, обеспечивающее прозрачную работу программ в неоднородной сетевой среде). В качестве открытого, нейтрального к поставщикам стандарта, LDAP обеспечивает наращиваемую архитектуру для централизованного хранения и управления информацией, которая должна быть доступной для современных распределенных систем и сервисов. LDAP в настоящее время поддерживается в большинстве сетевых операционных систем коллективного пользования и даже архивированных сетевых приложениях.

См. http://publibb.boulder.ibm.com/Redbooks.nsf/RedbookAbstracts/sg_244986.html?open.

Link Template™ (шаблон связи) - см. шаблон контекста.

Local Context (локальный контекст) - локальный контекст относится к информационным объектам и агентам клиентской стороны, доступных пользователям. Это включает агентов в семантической среде, локальные файлы, папки, элементы электронной почты в пользовательских почтовых ящиках, пользовательские любимые и последние Web-страницы, текущие Web-страницы, текущие открытые документы и другие информационные объекты, которые представляют текущие задачи пользователя, место, время или условие.

Значение (смысл) - признаки (атрибуты) линии поведения информации, которая позволяет потребителю информации определять ее местонахождение и перемещаться к ней, основываясь на содержании релевантной информации (в противоположность ее тексту или данным) и действовать на нее контекстным и зависимым от времени образом, чтобы максимизировать полезность информации.

Метаданные - «данные о данных». Они включают в себя те поля данных, связи и признаки, которые полностью описывают информационный объект.

Синтаксический анализатор естественного языка - компонент программного обеспечения синтаксического анализа и интерпретации, который понимает запросы на естественном языке и может преобразовать их в структурированные запросы семантической информации.

Nervana™ - зарегистрированное имя специализированной сквозной реализации среды/платформы информации информационной нервной системы. Это имя также определяет специализированное пространство имен для описателей (спецификаторов) типа ресурсов и предикатных имен.

.NET Passport - Microsoft .NET Passport является набором Web-сервисов, направленных на Интернет- и онлайн-покупки. .NET Passport обеспечивает пользователей функциональными возможностями единой регистрации (SSI) и быстрых покупок во все более возрастающем количестве участвующих сайтов, сокращая объем информации, которую пользователи должны запоминать или набирать при вводе. .NET Passport обеспечивает высококачественный онлайн-опыт для большой пользовательской базы и использует эффективные технологии шифрования, такие как SSL (протокол защищенных сокетов) и алгоритм 3DES (стандарт тройного шифрования данных) для защиты данных. Секретность является ключевым приоритетом, и все участвующие стороны подписывают контракт, в котором они выражают согласие сообщать и следовать стратегии секретности, которая придерживается промышленно принятых норм.

Сетевые эффекты - это имеет место, когда ряд других пользователей влияют на стоимость продукта или услуги для конкретного пользователя. Телефонные услуги дают ясный пример этого. Стоимость телефонного обслуживания для пользователей является функцией числа других абонентов. Немногие были бы заинтересованы в телефонах, которые не были бы соединены с любым абонентом, и большинство признали бы более высокую стоимость за телефонной услугой, подсоединенной к национальной сети, в противоположность локальной сети. Аналогичным образом, многие компьютерные пользователи высоко ценят компьютерную систему, которая позволяет им легко обмениваться информацией с другими пользователями.

Сетевые эффекты, таким образом, являются проявлениями потребностей, которые создают позитивный эффект обратной связи, в котором успешные продукты становятся более успешными. Таким образом, сетевые эффекты аналогичны мерам экономии со стороны поставщика в отношении масштаба и объема. По мере увеличения выходной продукции фирмы, меры экономии в связи с масштабом производства ведут к снижению средних затрат, позволяя фирме понизить цены и привлечь на свою сторону дополнительный бизнес от конкурентов. Непрерывное расширение приводит к снижающимся средним затратам, обосновывая все большее снижение цен. Аналогичным образом, позитивная обратная связь от сетевых эффектов строится на предыдущем успехе. В компьютерной индустрии, например, пользователи платят больше за более популярную компьютерную систему, во всем остальном одинаковую с другими, или за систему с большей базой установки, если цены и другие признаки двух конкурирующих систем эквивалентны. См. <http://www.ei.com/publications./1996/fall1.htm>.

Network News Transfer Protocol (NNTP) - (сетевой протокол передачи новостей) - протокол для рас-

пространения, запроса, поиска и посылки новых статей с использованием надежной потоковой передачи новостей сообществу ARPA (управление перспективных исследовательских программ США)-Интернет. Протокол NNTP создан таким образом, что статьи новостей сохраняются в центральной базе данных, позволяя абонентам выбрать только те элементы, которые им желательно прочитать. Индексация, перекрестные ссылки и удаление устаревших сообщений также обеспечиваются.

Уведомления - уведомления посылаются информационным агентом или агентством для указания пользователю, что у агента (как у агента клиентской стороны, так и у агента серверной стороны) имеется новая информация. Пользователи могут запросить уведомлений от агента в их семантической среде. Пользователи могут указать, что они получили уведомление. Источник уведомления (клиент или сервер) сохраняет информацию для пользователя, и агент указывает последнее время, когда пользователь подтвердил уведомление для агента. Источник уведомления опрашивает агента для проверки, имеется ли новая информация с времени последнего подтверждения. Если есть, то источник уведомления уведомляет пользователя. Уведомления могут быть посланы по электронной почте, на пейджер, в голосовом режиме, или с использованием специального механизма уведомления, такого как услуга .NER AlertsSM предлагаемая компанией Microsoft. Пользователи могут выбрать вариант указания предпочтительного для них механизма уведомления для всех источников уведомления (клиент или сервер), который применим для всех агентов на основе для источника уведомления или на основе для агента (что переопределяет указанное предпочтение источника уведомления).

Объект - см. информационный объект.

Тип объекта - идентификационные данные, связанные с информацией, которая позволяет потребителю понять сущность информации, интерпретировать ее содержание, прогнозировать, каким образом на информацию можно воздействовать, и связать ее с другими релевантными элементами информации на основе того, каким образом типы объектов в типовом случае связаны в реальном мире. Примеры включают документы, события, сообщения электронной почты, людей и т.д.

Онтология - иерархическое структурирование знаний в соответствии с существенными качествами. Онтология есть явное определение концептуализации. Термин позаимствован из философии, где «онтология» есть систематическая информация существования (наличия). Для систем искусственного интеллекта «существует» то, что может быть представлено. Если знания предметной области представлены в форме декларативного формализма, множество объектов, которые могут быть представлены, называется универсумом дискурса. Это множество объектов и описываемые соотношения между ними отражаются в репрезентативном словаре, которым основанная на знания программа представляет знания. Таким образом, в контексте искусственного интеллекта, онтология программы описывается определением множества репрезентативных терминов. В такой онтологии определения связывают имена сущностей в универсум дискурса (например, классы, отношения, функции и другие объекты) с читаемым человеком текстом, описывающим, что означают имена, и формальными аксиомами, которые ограничивают интерпретацию и точно сформулированное использование этих терминов. Формально, онтология есть положение логической теории.

Субъект онтологии есть исследование категорий вещей, которые существуют или могут существовать в некоторой предметной области. Продукт такого исследования, называемый онтологией, есть каталог типов вещей, которые подразумеваются существующим в интересующей предметной области D, исходя из перспективы персоны, которая использует язык L в целях обсуждения D. Типы в онтологии представляют предикаты, смысл слов, понятие и типы отношений языка L при использовании для обсуждения предметов в области D. См. <http://www-ksl.Stanford.edu/kst/what-is-an-ontology.html> и <http://users.bestweb.net/~sowa/ontology/>.

Предикаты - предикат есть признак или связь, результат которого представляет истинность или ложность некоторого условия. Например, предикат "authored by" ("создано автором" является ...) связывает персону с информационным объектом и указывает, является ли персона автором объекта.

PresenterTM (презентатор) - системный компонент в информационном агенте (семантический браузер) настоящего изобретения, который обрабатывает агрегирование и представление результатов с процессора семантических запросов (который предпочтительно интерпретирует SQML). Презентатор обрабатывает управление компоновкой (структурированием), агрегирование, навигацию, управление «поверхностью» (оболочкой), представление контекстных палитр, интерактивность, анимацию и т.д.

RDF - стандарт консорциума WWW на описание ресурсов - есть основа для обработки метаданных, он обеспечивает взаимодействие между приложениями, которые обмениваются понятной для машин информацией по сети Web. RDF определяет средства для обеспечения автоматизированной обработки Web-ресурсов. RDF определяет простую модель для описания соотношений между ресурсами в терминах именованных свойств и значений. RDF свойства можно представить как атрибуты ресурсов, и в этом смысле они соответствуют традиционным парам «атрибут (признак)-значение». Свойства RDF также представляют соотношения между ресурсами. Как таковая, модель данных RDF может поэтому напоминать диаграмму «сущность-соотношение».

RDF может использоваться в различных областях приложений, включая, например, обнаружение

ресурса для обеспечения лучших возможностей поискового процессора, каталогизация для описания содержимого и связей содержимого, доступных на конкретном Web-сайте, странице или цифровой библиотеке, интеллектуальными программными агентами для облечения совместного использования и обмена знаниями, для оценки содержимого, описания, набора страниц, которые представляют единый логический «документ», для описания прав интеллектуальной собственности Web-страниц, для выражения частных предпочтений пользователя, а также стратегии конфиденциальности Web-сайта. RDF с цифровыми подписями является предпочтительно компонентом построения "Web-сети доверия" для электронной коммерции, сотрудничества и других приложений. См. <http://www.w3.org/TR/PR-rdf-syntax/> и <http://www.w3.org/TR/rdf-schema/>.

RDFS - сокращение для схемы RDF. Специалистам по описанию ресурсов требуется возможность высказать некоторые сведения об определенных видах ресурсов. Для описания библиографических ресурсов, например, общими являются следующие описательные атрибуты: «автор», «название», «предмет». Для цифровой сертификации часто требуются такие атрибуты, как «контрольная сумма», «авторизация». Декларирование этих свойств (атрибутов) и их соответствующая семантика определены в контексте RDF как схема RDF. Схема определяет не только свойства ресурса (например, название, автор, предмет, размер, цвет и т.д.), но и также может определять виды описываемых ресурсов (книги, Web-страницы, люди, компании и т.д.). См. <http://www.w3.org/TR/rdf-schema/>.

Results Pane™ (панель результатов) - зарегистрированное имя для области графического отображения в пределах информационного агента (семантический браузер), которая отображает результаты запроса SQL. См. фиг. 5, показывающую вид экрана информационного агента, иллюстрирующий агентов серверной стороны, инструментальную панель управления воспроизведением/навигации/фильтрации, «диалог агентов серверной стороны» (который позволяет пользователям просматривать и открывать агентов серверной стороны), и производить выборку результатов (с типом информационного объекта «документы») из агента серверной стороны.

Семантика - коннотативное значение.

Semantic Environment™ (семантическая среда) - это понятие относится ко всем данным, сохраненным на локальных машинах пользователей, в дополнение к специфическим пользовательским данным на сервере агентства (например, абонированные агентства серверной стороны, любимые агенты серверной стороны и т.д.). Состояние клиентской стороны включает в себя любимого и последнего агентов и информацию аутентификации и авторизации (например, имена и пароли пользователя для разных агентств), в дополнение в SQL-файлам и буфера для каждого агента клиентской стороны (созданного пользователем). Информационный агент предпочтительно конфигурируется для сохранения агентов в течение установленного времени перед автоматическим их удалением, за исключением тех, которые внесены в список «любимых». Например, пользователи могут конфигурировать информационного агента для сохранения агентов в течение двух недель. В этом случае агенты старше двух недель автоматически удаляются из системы и семантическая среда настраивается соответствующим образом. Семантическая среда используется для контекстных палитр (контекстные палитры используют агентства в списка «последних» или «любимых», чтобы прогнозировать, из каких установленных по умолчанию агентств пользователи пожелают просмотреть контекст).

Semantic Environment Manager™ (администратор семантической среды) - зарегистрированное имя компонента программного обеспечения, который управляет всем локальным состоянием семантической среды (в информационном агенте). Это включает в себя сохранение и управление метаданными для всех агентов клиентской стороны (и предысторию и подписки любимых агентов), состояние на каждого агента (например, поверхности (оболочки) агента, предпочтения агента и т.д.), управление уведомлениями, просмотр ресурсов агентства (по каталогам агентства), прослушивание агентств по протоколам многоадресных передач одноранговых уведомлений, сервисы, обеспечивающие возможность пользователям просматривать семантическую среду посредством семантического браузера (посредством просмотра по дереву, диалога «открыть агента» и панели результатов) и т.д.

Semantic Data Gatherer™ (SDG) (сборщик семантических данных) - зарегистрированное имя для XML-Web-сервиса, используемого сервером интеграции знаний (СИЗ), ответственного за пополнение, удаление и обновление записей в семантической сети посредством компонента памяти метаданных (SMS).

Semantic Metadata Store™ (SMS) (память метаданных) - зарегистрированное имя для компонента программного обеспечения в СИЗ, который использует базу данных (например, SQL Server, Oracle, DB2), имеющую таблицы для каждого основного типа объекта для хранения всех метаданных в СИЗ.

Семантическая сеть - система и способ связывания объектов, связанных со схемами систематическим образом посредством таблиц баз данных в памяти семантических метаданных.

Semantic Network Consistency Checker™ (компонент проверки согласованности семантической сети) - зарегистрированное имя для компонента программного обеспечения, который выполняется на агентстве согласно настоящему изобретению, задачей которого является поддержание целостности и согласованности семантической сети. Компонент проверки выполняется периодически и гарантирует, что записи в

таблице «семантические связи» существуют в таблицах «родных» (собственных, присущих среде) объектов, что записи в таблице «объекты» существуют в таблицах собственных объектов и что все записи в памяти семантических метаданных все еще существуют в хранилищах, из которых они были собраны.

Семантические запросы - запросы, которые воплощают в себе значение (смысл), контекст, зависимость от времени, контекстные шаблоны, содержательность, приближающуюся к естественному языку. Намного более мощные, чем простые, основанные на ключевых словах запросы, так как они являются контекстно- и времязависимыми и содержат в себе смысл и семантику.

Semantic Query Markup Language (SQML) - (язык разметки семантических запросов) - специализированный, основанный на XML язык запросов, используемый в данном изобретении, для определения, сохранения, интерпретации и исполнения семантических запросов клиентской стороны. SQML включает в себя метки для определения запроса, который получает свои данные из различных ресурсов (которые представляют источники данных), таких как файлы, папки, хранилища приложений и ссылки на XML-Web-сервисы агентств (посредством идентификаторов ресурсов и URL). Кроме того, SQML включает в себя метки, которые обеспечивают семантические фильтрации (посредством настроенных связей и предикатов), которые указывают, каким образом данные должны запрашиваться и отфильтровываться из ресурсов, и аргументы, которые указывают, каким образом ресурсы должны запрашиваться, и каким образом должны отфильтровываться результаты. В частности, аргументы могут включать в себя ссылки на локальный или удаленный контекст. Контекстные аргументы затем превращаются посредством SQP (процессора семантических запросов) клиентской стороны во время исполнения в XML-метаданные. XML-метаданные затем пропускаются к соответствующему ресурсу (например, XML-Web-сервису агентства) как вызов метода вместе со ссылкой на ресурс и семантическими связями и предикатам, которые указывают, каким образом запрос должен быть преобразован ресурсом (например, XML-Web-сервисом агентства). SQML по отношению к информационной нервной системе играет ту же роль, что и HTML для современной Web-сети. Основное различие состоит в том, что SQML определяет правила для семантических запросов, в то время как HTML определяет правила для гипертекстового представления. Однако SQML обладает преимуществами в том, что он позволяет клиенту рекурсивно создавать новые семантические запросы из существующих (путем создания нового SQML-запроса с новыми связями, выведенного из существующего SQML-запроса), например, посредством операций графического интерфейса «перетащить и оставить» и интеллектуального копирования и вставки, интеллектуальной лупы, шаблонов и палитр контекста и т.д. Кроме того, поскольку SQML не определяет правил для представления, результаты семантического запроса могут быть представлены множеством путей, используя «поверхность», которая применяет результаты (в SRML) для генерации представления на основе пользовательских предпочтений, интересов, условия или контекста. Кроме того, SQML может содержать абстрактные связи и предикаты, такие как те, которые ссылаются или используют шаблоны контекста. Ресурс (например, XML-Web-сервис агентства) затем преобразует SQML в соответствующий формат запроса (например, SQL или эквивалентный ему, в случае XML-Web-сервиса агентства) и затем вызывает «действительный» запрос для генерации результатов (которые затем будут вычислять контекст пользователя или шаблон контекста). Также, SQML-буфер или файл может ссылаться на множество ресурсов (и агентств), тем самым позволяя пользователю просматривать результаты агрегированным способом (например, на основе контекста или во временной зависимости), а не на основе источника данных, - это особенно эффективный признак изобретения, который обеспечивает управляемый пользователем просмотр ресурсов и агрегирование информации (см. разделы по обоим вопросам, приведенные ниже). Наконец, каждый агент клиентской стороны имеет SQML-определение и файл, точно так, как каждая Web-страница имеет HTML-файл.

Semantic Query Processor™ (SQP) (процессор семантических запросов) - зарегистрированное имя для процессора семантических запросов серверной стороны (XML-Web-сервис в предпочтительном варианте осуществления), который принимает SQML-запрос и преобразует его в SQL (в предпочтительном варианте) и затем возвращает результаты как XML. На сервере интеграции знаний процессор семантических запросов является главной точкой входа в семантическую сеть согласно настоящему изобретению, и несет ответственность за реагирование на семантические запросы от клиентов сервера интеграции знаний. На сервере это компонент программного обеспечения, который обрабатывает семантические запросы, представленные как SQML, от клиента. На клиентской стороне процессор семантических запросов принимает агрегированный SQML и компилирует или отображает его на индивидуальные SQML-запросы, которые могут быть пересланы на XML-Web-сервис сервера (или агентства).

Semantic Results Markup Language (SRML) (язык разметки семантических результатов) - специализированная основанная на XML схема данных и формат, используемые настоящим изобретением для определения, хранения, интерпретации представления семантических результатов. У клиента, SRML возвращается от процессора семантических запросов через обработчики семантических ресурсов, которые интерпретируют, форматируют и выдают требования запросов на источники семантических данных. Источники семантических данных будут включать в себя XML-Web-сервис агентства, локальные файлы, локальные папки, заказные источники данных от локальных и удаленных приложений (например, почтовый ящик электронной почты Microsoft Outlook) и т.д. XML-Web-сервис будет возвращать SRML клиен-

ту в ответ на семантический запрос клиента. Таким образом, XML-Web-сервис не будет «заботиться» о том, каким образом результаты представляются клиенту. Это контрастирует с современной Web-сетью и семантической Web-сетью, где серверы возвращают уже отформатированный HTML клиенту для представления и где клиенты просто представляют данные представления (в противоположность семантическим данным) и не могут настраивать представление данных. В настоящем изобретении два клиента могут воспроизвести один и тот же SRML полностью различными путями, основываясь на текущей «поверхности», которая была выбрана или применена пользователем каждого клиента. «Поверхность» затем преобразует SRML в готовый для представления формат, такой как XHTML, DHTML+TIME, SVG, Flash MX и т.д.

SMRL является метасхемой, что означает, что это является форматом контейнера, который может включать в себя данные для различных типов информационных объектов (например, документы, электронная почта, люди, события и т.д.). SRML-файл или буфер может содержать промежуточные результаты для каждого из этих типов объектов. Правильно построенный SRML должен содержать правильно построенные разделы XML-документа, которые согласованы со схемой типов информационных объектов, которые содержатся в семантическом результате, который представляет SRML. См. образец А в приложении к настоящему описанию.

Семантическая сеть Web - расширение современной сети Web, в котором информации придано правильно определенное значение, позволяющее компьютерам работать во взаимодействии более эффективно. См. Tim Berners-Lee, James Hendler, Ora Lassila, The Semantic Web, Scientific American, May 2000.

Средства для введения понимаемых машинами данных в современную сеть Web становятся высокоприоритетными для многих сообществ. Сеть Web может достичь своего полного потенциала, только если она станет местом, где данные могут совместно использоваться, обрабатываться автоматическими средствами, а также людьми. Для такой сети Web будущие программы должны обеспечивать возможность совместно использовать и обрабатывать данные, даже если эти программы созданы полностью независимым образом. Семантическая сеть Web является концептуальным мировоззрением: идея наличия в сети Web данных, определенных и связанных таким путем, что они могут быть использованы машинами не только в целях отображения, но для автоматизации, интеграции и повторного использования данных в самых различных приложениях. См. также <http://www.w3.org/2001/sw/>.

Session Announcement Protocol (SAP) (протокол извещения о сеансе). Чтобы оказать помощь в извещении о многоадресных мультимедийных конференциях и других многоадресных сеансах, и для передачи релевантной информации настройки сеанса потенциальным участникам, может быть использован распределенный каталог сеанса. Экземпляр такого каталога сеанса периодически транслирует пакеты, содержащие описание сеанса, и эти извещения принимаются другими каталогами сеанса, так что потенциальные удаленные участники могут использовать описание сеанса для запуска инструментальных средств, требуемых для участия в сеансе.

В своей простейшей форме это связано с периодической многоадресной передачей пакета извещения о сеансе, описывающего конкретный сеанс. Для приема извещения SAP приемник просто прослушивает хорошо известный адрес и порт многоадресной передачи. Сеансы описываются с использованием протокола описания сеанса (<ftp://ftp.isi.edu/in-notes/rfc2327.txt>). Если приемник принимает пакет извещения о сеансе, он просто декодирует сообщение SDP и затем может отобразить информацию о сеансе для пользователя. Интервал между повторениями сообщения с описанием одного и того же сеанса зависит от числа сеансов, о которых извещается (каждый отправитель в конкретной области действия может слышать других отправителей в той же самой области действия), так что ширина полосы, используемая для извещений о сеансе в конкретной области, может поддерживаться примерно постоянной. Если приемник осуществлял прослушивание в течение некоторого времени и не услышал извещения о сеансе, то приемник может заключить, что сеанс отменен и больше не существует. Установленный интервал основывается на оценке приемника того, как часто отправитель должен осуществлять передачу. См. <http://www.fags.org/rfc2974.html>, http://www.video.ja.net/mice/archive/sdr_docs/nodel.html, <ftp://ftp.isi.edu.in-notes/rfc2327.txt>.

Simple Mail Transfer Protocol (SMTP) (простой протокол электронной почты) - протокол, предназначенный для надежной и эффективной пересылки электронной почты. SMTP не зависит от конкретной подсистемы пересылки и требует только надежного потокового канала упорядоченных данных. Важной особенностью SMTP является его способность транслировать почту в транспортной среде. См. <http://www.ietf.org/rfc/rfc0821.txt>.

Поверхности (оболочки) - шаблоны представления, которые используются для настройки пользовательского опыта на основе по каждому агенту или для настройки всей структуры (формата) (независимо от агента), или объекта (на основе типа информационного объекта), контекста (на основе шаблона контекста), элемента сопряжения (для агентов, являющихся элементами сопряжения) для имени/пути семантической области или онтологии и других факторов. Каждый агент должен включать в себя поверхность, которая в свою очередь должна иметь представление XML-метаданных для параметров, чтобы настраивать структуру (топологию) XML-результатов, которые представляют информационные объекты (по-

верхность топологии), например, должны ли быть эти результаты анимированными, и способ, которым отображается каждый результат, включая представление типа объекта (поверхность объекта), стили, цвет, графику, фильтры, трансформации, эффекты, анимации и т.д., что указывает онтологию текущих результатов (поверхность онтологии), стили, которые указывают шаблон контекста текущих результатов (поверхность контекста), и стили, которые указывают, как просматривать и перемещаться по результатам из элементов сопряжения (т.е. поверхность элемента сопряжения).

Smart Lens™ (интеллектуальная лупа) - зарегистрированное имя специализированной функции данного изобретения, позволяющей пользователю выбирать интеллектуального агента или объект в качестве контекста, с которым просматривают другой объект, или агента. Лупа отображает метаданные, связи и результирующие предварительные просмотры, которые дают пользователям указание того, что им следует ожидать, если контекст будет вызван. По существу, интеллектуальная лупа отображает результаты «потенциального запроса». Интеллектуальная лупа позволяет пользователям быстро просматривать результаты контекста без действительного вызова запросов (тем самым увеличивая их производительность). Кроме того, интеллектуальная лупа может отображать виды, которые согласованы с контекстом, использовать опорные пункты, шаблоны и окна предварительного просмотра, позволяя пользователю анализировать контекст различными путями перед вызовом запроса.

Smart Virtual Web™ (интеллектуальная виртуальная сеть Web) - зарегистрированное имя для свойства согласно настоящему изобретению, заключающегося в интеграции семантики, контекстной зависимости, зависимости от времени и динамики для обеспечения возможности пользователям просматривать ресурсы динамической, виртуальной, оперативной, управляемой пользователем сети Web, позволяя им осуществлять управление и настройку. Это находится в контрасте с современной сетью Web и концептуальной семантической сетью Web, которые используют ручную создаваемую сеть, в которой пользователи зависят от создателей информации в сети.

Structured Query Language (SQL) - (структурированный язык запросов) - используется для информационного обмена с базой данных. В соответствии с ANSI (Американский Национальный институт стандартов), это есть стандартный язык для систем управления реляционными базами данных. SQL-формулировки используются для выполнения таких задач, как обновление данных в базе данных или извлечение данных из базы данных. Некоторыми известными системами управления реляционными базами данных являются следующие: Oracle, Sybase, Microsoft SQL Server, Access, Ingress и т.д. Хотя большинство баз данных используют SQL, большинство из них также обладают собственными дополнительными расширениями, которые обычно используются только в их системе. Однако стандартные SQL-команды, такие как «выбрать», «вставить», «обновить», «удалить», «создать», «сбросить», могут быть использованы для выполнения практически всего, что требуется при обращении с базой данных.

SQL работает с реляционными базами данных. Реляционная база данных сохраняет данные в таблицах (соотношениях, связях). База данных есть набор таблиц. Таблицы состоят из списка записей, каждая запись в таблице предпочтительно включает в себя одну и ту же структуру, и каждая имеет фиксированное число «полей» заданного типа. См. <http://www.sqlcourse.com/info.html> и <http://www.dcs.napier.ac.Uk/~andrew/sql/0/w.htm>.

Scalable Vector Graphics (SVG) (масштабируемая векторная графика) - язык для описания двумерной графики в XML. SVG учитывает три типа графических объектов: формы векторной графики (например, трассы, состоящие из прямых линий и кривых), изображение и текст. Графические объекты могут группироваться, изменяться по стилю, трансформироваться и разлагаться на предварительно визуализированные объекты. Текст может быть в любом пространстве имен, пригодном для приложения, которое улучшает возможности поиска и доступность для SVG графики. Набор признаков включает в себя вложенные трансформации, отсечение областей, альфа-маски, фильтровые эффекты, объекты шаблонов и наращиваемость. SVG-рисунки могут быть динамическими и интерактивными. Объектная модель документа (DOM) для SVG, которая включает в себя полную модель XML DOM, позволяет реализовать простую и эффективную векторную графическую анимацию посредством написания сценариев. Богатый набор обработчиков событий, таких как onmouseover и onclick, могут быть назначены для любого SVG-графического объекта. Ввиду совместимости и отношения поддержки с другими Web-стандартами, функции, подобные созданию сценариев, могут быть реализованы на SVG-элементах и других XML-элементах из различных пространств имен одновременно в пределах одной и той же Web-страницы. См. <http://www.w3.org/Graphics/SVG/Overview.htm#>.

Таксономия (систематика) - организационная структура, в которой подразделения упорядочены в группы или по категориям.

Временная зависимость - свойство информационной среды доставлять и представлять информацию на основе того, когда информация была бы наиболее релевантной по времени. Например, «свежесть» (новизна) является признаком, обозначающим временную зависимость. Кроме того, доставка и представление наступающих событий (которые по определению являются зависимыми от времени) и способ, которым критичные ко времени события отображаются, являются свойствами зависимой от времени среды.

Современная сеть Web - это определение относится к общеизвестной на сегодняшний

момент Всемирной Паутине. Современная сеть Web является универсумом гипертекстовых серверов (HTTP серверов), которые представляют собой серверы, позволяющие связать между собой текст, графику, звуковые файлы и т.д. Гипертекст - есть просто нелинейный способ представления информации. Вместо того, чтобы читать или изучать некоторые предметы в порядке, установленном для нас автором, или редактором, или издателем (сервером публикаций), читатели гипертекста могут следовать своим собственным путем, создают свой собственный порядок или придают значение материалу. Это выполняется путем создания «связей» между информацией. Эти связи обеспечиваются так, что пользователь может «перескакивать» к следующей информации о конкретном обсуждаемом вопросе (который может иметь больше связей, уводящих читателя в другое направление). Гипертекстовая среда может включать в себя картинки, звуки, виды, присутствующие в мультимедийном методе представления информации, также называемом гипермедийным. См. <http://www.w3.org/History.html> и <http://www.umassd.edu/Public/people/KAmaral/Thesis/hypertext.html>.

Multicast Time to Live (TTL) (время жизни многоадресной передачи) - протокол маршрутизации многоадресной передачи использует поле дейтаграмм для принятия решения, насколько «далеко» от посылающего хоста должен пересылаться данный многоадресный пакет. Установленный по умолчанию TTL для многоадресных дейтаграмм есть 1, результатом чего является то, что многоадресные пакеты поступают только к другим хостам в локальной сети. Опция `setsockopt(2)` может быть использована для изменения TTL. По мере того как значение для TTL возрастает, маршрутизаторы будут увеличивать количество транзитных участков («прыжков»), на которые они должны переслать многоадресный пакет. Для обеспечения осмысленного контроля области действия, многоадресные маршрутизаторы в типовом случае используют следующие «пороги» при пересылке на основе TTL:

- 0 - ограничено тем же хостом
- 1 - ограничено той же подсетью
- 32 - ограничено тем же сайтом
- 64 - ограничено тем же регионом
- 128 - ограничено тем же континентом
- 255 - без ограничений.

См. <http://www.isl.org/projects/eies/mbone/mbone27.htm>.

Пользовательское состояние - относится ко всему состоянию, которое либо создано пользователем, либо необходимо для сохранения пользовательских предпочтений, любимых вариантов выбора или любой другой персональной информации на клиентской или серверной стороне. Пользовательское состояние на клиентской стороне включает в себя информацию удостоверяющих данных аутентификации, список агентов пользователя (и все метаданные, включая SQL-запросы для агентов), исходного агента, опции конфигурации, предпочтения, такие как «поверхности» и т.д. По существу, пользовательское состояние на клиентской стороне есть постоянная форма пользовательской семантической среды. Пользовательское состояние на серверной стороне включает в себя такую информацию, как пользовательские любимые агенты, абонированные агенты, установленные по умолчанию агенты, семантические связи с информационными объектами на сервере (например, связи с «любимыми» объектами) и т.д.

Пользовательское состояние на серверной стороне является опционным для серверов, но его поддержка является предпочтительной. Серверы обычно поддерживают вход в систему (регистрацию) пользователя и тип объекта «люди» (даже без агентов серверной стороны), поскольку они нужны для таких свойств, как любимые объекты, рекомендации и для шаблонов контекста, таких как «нюсмейкеры», «эксперты», «рекомендации», «любимые», «классические».

Virtual Information Object Type™ (тип виртуального информационного объекта) - зарегистрированное имя для типов объектов, которые не отображаются на отдельные типы объектов, но семантически представляют интерес для пользователей.

Virtual Parameter™ (виртуальный параметр) - зарегистрированное имя для переменных, параметров, аргументов или имен, которые динамически интерпретируются во время исполнения процессором семантических запросов. Это позволяет администратору агентства сохранять агентов, которые ссылаются на виртуальные имена и затем преобразовывать эти имена в действительные релевантные термины, когда запрос вызывается.

Сеть Web Доверия - термин, придуманный членами сообщества исследователей семантической сети Web, который относится к цепочке авторизаций, которую должны использовать пользователи семантической сети Web для подтверждения утверждений (описывающих семантические свойства элементов) и формулировок. На основе трудов в математике и криптографии, цифровые подписи обеспечивают доказательство того, что некоторое лицо подписало (и выразило согласие) документ или формулировку. Пользователи могут предпочтительно в цифровом виде подписывать все свои RDF-утверждения. Таким образом, пользователи могут быть уверены, что они подписали их (или, по меньшей мере, ручаться за их аутентичность). Пользователи просто сообщают программе, чьим подписям доверять. Каждый может установить свои собственные уровни доверия, и компьютер может принимать решение о том, как много из того, что считано, является достоверным.

Например, с помощью сети Web-доверия пользователь может сообщить компьютеру, что он доверяет своему лучшему другу Роберту. Роберт, кажется, будет довольно популярной фигурой в сети, доверяющей порядочному числу людей. Все люди, которым он доверяет, в свою очередь, будут доверять другому множеству людей. Каждая из этих мер доверия имеет определенную степень (Роберт может доверить Венди многое, но Салли - лишь немного). Помимо доверия, уровни недоверия также могут быть факторизованы. Если пользовательский компьютер обнаруживает документ, которому явно никто не выразил доверия, но также ни один не сказал, что он полностью ложный, то он, вероятно, будет доверять этой информации несколько больше, чем той, по отношению к которой многие люди высказались, что она ложна. Компьютер принимает все эти факторы во внимание при принятии решения о достоверности элемента информации. Предпочтительно компьютер объединяет всю эту информацию в простое отображение (истинно-ложно) или в более сложное объяснение (описание всех различных имеющихся отношении факторов доверия). См. <http://blogspace.com/rdf/SwartzHendler>.

Web Services Interoperability (WS-I) (возможность взаимодействия Web-сервисов) - открытая организация, имеющая привилегии для того, чтобы способствовать продвижению взаимодействия Web-сервисов между платформами, операционными системами и языками программирования. Эта организация работает с промышленными организациями и с организациями по стандартизации для реагирования на пользовательские потребности путем обеспечения руководящих материалов, эффективной практики и ресурсов для разработки решений, связанных с Web-сервисами. См. <http://www.ws-i.org>.

Web Services Security (WS-Security) (защищенность Web-сервисов) - усовершенствования к SOAP-передаче сообщений, обеспечивающие качественную защиту через целостность сообщений, конфиденциальность сообщений и аутентификацию отдельных сообщений. Эти механизмы могут быть использованы для учета широкого разнообразия моделей защиты и технологий шифрования. WS-Security также обеспечивает универсальный механизм для связи маркеров доступа с сообщениями. Никакого особого типа маркера доступа не требуется для WS-Security. Он спроектирован для расширения (например, поддерживает множество форматов маркеров доступа). Например, клиент может предоставить доказательство подлинности и доказательство, что он имеет сертификацию определенного бизнеса. Дополнительно WS-Security описывает, как кодировать двоичные маркеры доступа. В частности, спецификация описывает, как кодировать X 509 сертификаты и Kerberos tickets («мандаты Цербера»), а также как включать непрозрачные зашифрованные ключи. Она также включает механизм наращиваемости, который может использоваться для дополнительного описания характеристик удостоверений личности (мандатов доступа), которые включаются в сообщение. См. <http://msdn.microsoft.com/library/default.asp?url=/libraryen-us/dnglobspec/html/ws-security.asp>.

Extensible Markup Language (XML) - (расширяемый язык разметки) - универсальный формат для структурированных документов и данных в сети Web. Структурированные данные включают в себя такие элементы, как крупноформатные таблицы, справочники, параметры конфигураций, финансовые транзакции, технические чертежи. XML есть набор правил (можно также представлять их как руководящие материалы или соглашения) для разработки текстовых форматов, которые позволяют вам структурировать ваши данные. XML не является языком программирования, и не требуется быть программистом, чтобы использовать или изучать его. XML упрощает возможность для компьютера генерировать данные, считывать данные и обеспечивать однозначность структуры данных. XML избегает обычных ловушек в проектировании языка: он является наращиваемым, независимым от платформы, поддерживает интернационализацию и локализацию XML, полностью совместим с Unicode. См. <http://www.w3.org/XML/1999/XML-in-10-point>.

XML Web Service (XML-Web-сервис, также известный как Web-сервис) - сервис, обеспечивающий стандартное средство информационного обмена между различными программными приложениями, участвующими в представлении динамической, управляемой контекстом информации для пользователя. Более конкретные определения включают следующее.

1. Прикладная программа, идентифицируемая посредством URL, для которой интерфейсы и связи могут быть определены, описаны и обнаружены XML-артефактами. Поддерживает прямые взаимодействия с другими прикладными программами, используя XML-сообщения по протоколам на основе Интернет.

2. Прикладная программа, доставляемая как сервис, который может быть интегрирован в любые Web-сервисы с использованием стандартов Интернет. Это ресурс, адресуемый с помощью URL, который программным образом возвращает клиентам информацию, которую они хотят использовать. Основным используемым коммуникационным протоколом является SOAP (протокол простого доступа к объекту), который в большинстве случаев представляет собой XML на HTTP.

3. Программируемая логика прикладных программ, доступная с использованием стандартных протоколов Интернет. Web-сервисы объединяют аспекты разработки на основе компонентов и Web. Подобно компонентам, Web-сервисы представляют функциональные способности черного ящика, которые могут повторно использоваться, не заботясь о том, как реализуется сервис. В отличие от текущих технологических компонентов, Web-сервисы не доступны через протоколы, специфические для моделей компонентов, такие как DCOM, RMI или HOP. Вместо этого, Web-сервисы доступны через повсеместно используемые

Web-протоколы (например, HTTP) и форматы данных (например, XML).

См. <http://www.xmlwebservices.cc/>, <http://www.perfectxml.com/WebSvc1.asp> и <http://www.w3.org/2002/ws/arch/2/06/wd-wsa-reqs-20020605.html>.

XQuery - язык очереди, который использует структуру XML для разумного формулирования запросов по всем видам данных, независимо от того, сохранены ли они физически на XML или рассматриваются как XML посредством связующего (промежуточного) программного обеспечения. См. <http://www.w3.org/TR/xquery/> и <http://www.com.developerworks/xml/library/x-xquery.html>.

XPath - результат попытки обеспечить общий синтаксис и общую семантику функциональными возможностями, используемыми совместно между XSL-трансформациями (<http://www.w3.org/TR/XSLT>) и X-указателем (<http://www.w3.org/TR/xpath#XPATH>). Основное назначение XPath - адресовать части XML[XML] документа. В поддержку этого основного назначения также обеспечиваются базовые средства для манипулирования строками, числами и булевыми операторами. XPath использует компактный, не XML-синтаксис для облегчения использования XPath среди значений с универсальными идентификаторами ресурсов (URI) и значениями признаков XML. XPath работает над абстрактной, логической структурой XML-документа, а не над его поверхностным синтаксисом. XPath получил свое имя от его использования в качестве указания пути как в URL для навигации в иерархической структуре XML-документа.

В дополнение к использованию для адресации, XPath также спроектирован так, что он имеет естественное подмножество, которое может быть использовано для согласования (проверки согласуется ли узел с шаблоном); это использование XPath описано в XSLT. XPath моделирует XML-документ как дерево узлов. Имеются различные типы узлов, включая узлы элементов, узлы признаков и узлы текстов. XPath определяет путь для вычисления значения строки (последовательности символов структуры данных) для каждого типа узла. Некоторые типы узлов также имеют имена. XPath полностью поддерживает пространство имен XML (<http://www.w3.org/TR/xpath#XMLNAMES>). Таким образом, имя узла моделируется как пара, состоящая из локальной части и, возможно, нулевого пространства имен URI; это называется <http://www.w3.org/TR/xpath#dt-expanded-name>. См. <http://www.w3.org/TR/xpath#XPATH>.

XSL - язык таблицы стилей для XML, который включает XML-словарь для определения форматирования. См. <http://www.w3.org/TR/xslt1/>.

XSLT - используется языком XSL для описания того, каким образом документ преобразуется в другой XML-документ, который использует словарь форматирования. См. <http://www.w3.org/TR/xslt1/>.

В. Обзор.

1. Контекст изобретения.

Бытует неправильное представление, что Святой Грааль для доступа к информации есть обеспечение возможностей поиска на натуральном языке. Ранние технологии для доступа к информации фокусировались главным образом на улучшении интерфейса для поиска или доступа к информации для оптимизации извлечения информации. Презумпцией в значительной степени был интерфейс натурального языка к информации, который совершенным образом должен был решить пользовательские проблемы доступа к информации и покончить с чувством разочарования, которые пользователи испытывают при поиске информации.

Однако в действительности многие направления анализа связаны с тем, каким образом люди приобретают знания в реальном мире. Одним примером является контекст. Имеется много вещей, которые люди знают только потому, что они находились в определенном месте и в определенное время. Если бы они не были в том месте в то время, то они не узнали бы о том, что в действительности известно, или, реально, могли не позаботиться о том, чтобы узнать. Наличие способности отыскивать то, что в настоящее время известно с использованием натурального языка, не помогает обнаруживать знания, связанные с тем конкретным временем и местом. Просто отсутствуют натуральные параметры, которые сформируют корректный запрос для извлечения желательной информации.

Загадка есть то, о чем лицо не может спросить, поскольку то, чего он еще может не знать, будет иметь значение только после наступившего факта. Иными словами, лицо не может сделать запрос о том, что оно не знает, что это ему не известно, или о том, что он не знает, что это ему будет желательно узнать. Контекстная зависимость, временная зависимость, обнаружимость, динамическое связывание, управляемый пользователем просмотр ресурсов, пользовательская «Семантическая Среда», гибкое представление, поверхности контекстов, признаки контекстов, палитры контекстов (что доставляет релевантную, зависимую от контекста и от времени информацию на основе шаблонов контекстов) и другие аспекты настоящего изобретения распознают и корректируют этот фундаментальный недостаток существующих информационных систем.

Например, люди могут иметь много CD в своей библиотеке (тем самым пополняя «знания» музыки), поскольку они посещали определенные вечера и беседовали с определенными людьми. Те люди на тех вечерах упоминали об этих CD в разговоре с данными людьми, тем самым способствуя расширению знаний данных людей о музыке. В качестве другого примера можно указать некоторое лицо, которое может купить книгу (которая, будучи прочитанной, расширит знания этого лица о конкретном предмете, рассматриваемом в книге), на основе рекомендации, полученной от знакомого до сих пор постороннего лица, с которым он случайно оказался рядом в самолете. В реальном мире люди приобретают знания

основываясь не на том, что они читают и отыскивают, а на основе общения с друзьями, взаимодействия с людьми, суждения людей, которым они доверяют. Эта «среда знаний», можно утверждать, настолько же критична, если не более критична для распространения и приобретения знаний, как и модель для поиска (цифровая или аналоговая).

Настоящее изобретение отражает виртуально сценарий приобретения знаний в реальном мире. Результирующая Information Nervous System™ (Информационная Нервная Система) является средой, реализующей большую часть работы, кроме сценариев, отображающих очень точно на аналоговый (реальный) мир.

Неспособность попыток методов поиска на естественном языке в современной сети Web, а также семантической сети Web для распознавания множества путей распространения и приобретения знаний делают их в конечном счете неэффективными. Настоящее изобретение рассматривает различные пути, которыми люди уже приобретали знания, независимо от реальной технологии, используемой в доставке информации.

К примеру, всегда имеется контекст и всегда имеет место время. Аналогичным образом, всегда имеется представление об обнаружении и необходимость связать информацию динамически и с пользовательским управлением. Всегда имелись определенные контексты шаблонов, хотя и в средах, отличающихся от представленных здесь, включая «классику», «предысторию», «временную шкалу», «наступающие события», «заголовки». Эти шаблоны существовали до создания сети Интернет, современной сети Web, электронной почты, электронного обучения и т.д. Тем не менее, до настоящего изобретения не было возможности в электронной среде сфокусироваться на режиме, протоколе и представлении доставки знаний, которая отображает сценарии реального мира (например, посредством шаблонов контекстов, контекстной зависимости, временной зависимости, динамического связывания, гибкого представления, поверхностей контекстов, признаков контекстов и т.д.) в противоположность реальным типам информации, семантическим связям, метаданным и т.д. Всегда будут иметься новые типы информации. Однако направления распространения и приобретения знаний (например, шаблоны контекстов) всегда были и будут оставаться теми же самыми. Настоящее изобретение основывается на этой реальности.

Кроме того, настоящее изобретение обеспечивает возможность распространения знаний через интуитивную прозорливость. Интуитивная прозорливость играет большую роль в приобретении знаний в реальном мире и является первоклассным режимом доставки знаний. Настоящее изобретение позволяет пользователю собирать информацию интуитивно (хотя и интеллектуальным образом), посредством ее поддержки контекстом, временем, шаблонами контекста и т.д.

Информационные модели или среды, которые используют строгую, статичную структуру, подобную "Web", разрушаются. Так как они предполагают наличие созданной (создателями, авторами) сети и не учитывают различных направлений формирования знаний. Такие информационные модели не нацелены на пользователя, не включают в себя контекст, время, динамику и шаблоны, и не отражают сценарии приобретения и распространения знаний в реальном мире. Настоящее изобретение минимизирует потери информации и максимизирует сохранение информации, даже в отсутствие сети Web как таковой, и даже если естественный язык не используется при поиске информации. Это возможно потому, что в отличие от существующих сред для доступа к информации, предпочтительный вариант осуществления изобретения фокусируется на моделях распространения знаний, которые включают в себя контекст, время, динамику, шаблоны (с выгодой для конечного пользователя и для производителя содержания), а не на специфических особенностях интерфейса доступа или на связывании (семантическом или несемантическом) ресурсов информации на основе статических моделей данных или авторских разработках. Во многих сценариях сеть Web (семантическая или несемантическая) необходима как средство навигации, но далеко не достаточна как средство распространения и приобретения знаний. Информационная нервная система согласно настоящему изобретению воплощает «направления знания», описываемые в изобретении (включая, но не ограничиваясь навигацией на основе связей), и интеллектуальным и интегрированным способом объединяет их для обеспечения распространения и приобретения знаний с выгодой для всех сторон, вовлеченных в процесс переноса знаний.

2. Значения задач.

В настоящее время знания должны быть вручную постоянно кодированы в цифровую среду информационной структуры, независимо от того, предназначаются ли они для предприятия, потребителя или для обычного запрашивающего населения. Если они не представлены создателями и не распространены надлежащим образом, то никто не будет знать об их существовании, об их связи с другими источниками информации и каким образом воздействовать на них в реальном времени и надлежащим образом. В значительной степени это объясняется тем, что современная сеть Web не проектировалась как платформа для знаний. Она проектировалась как платформа для представлений и поэтому по определению является «тупой», статичной и реактивной. В настоящее время специалисты в области информационных технологий, которые нацелены на использование информации за счет добавления контекста и смысла, оказываются в зависимости от авторов знаний.

Важным аспектом во взаимодействии со знаниями является обеспечение возможности специалистам в области информационных технологий перемещаться собственным путем в пространстве знаний

весьма интуитивным способом и со скоростью, которая обеспечит им принятие решений и воздействие на знания. Иными словами, специалисты в области информационных технологий не должны думать об «островке» электронного обучения, как отдельного от документов и организаций, сообщений электронной почты, содержащих обратную связь потребителей, медийных файлов, наступающих видеоконференций, встреч, состоявшихся недавно, информации, сохраненной в новостных группах или в относящихся к ней книгах. Предпочтительной ситуацией является отнести к низшему уровню «типы» и «источники» информации и создать «бесшовный (непрерывный) опыт на основе знаний», который противоположен всем таким островкам на семантической основе.

При создании опыта, основанного на знаниях, также предпочтительно имеет возможность интегрировать знания среди поставщиков содержимого, партнеров, изготовителей, потребителей и населения. В сценарии предприятия, например, ни одна организация не имеет полного знания, которое ей необходимо, чтобы быть конкурентно способной. Знания сохраняются в отчетах промышленности, исследовательских документах консалтинговых фирм и инвестиционных банков, медиа-фирм, подобных Reuters™, Bloomberg™ и т.д. Все это составляющие знаний. Недостаточно развернуть хранилище электронного обучения для обучения пользователей на одновременной или периодической основе. Пользователи должны иметь постоянный доступ к знаниям из различных источников, на месте, в интеллектуальном контексте, который релевантен их текущей задаче.

Все это требует уровня интеллектуальности и проактивности, которого нет в настоящее время. В настоящее время предприятия используют информационные порталы, такие как интранеты и Интернет в качестве пути распространения информации к своим служащим. Однако это далеко недостаточно, поскольку обеспечивает только интеграцию на уровне представления. Это сродни подписке на газеты для поддержки обновления информацией в противоположность использованию агента, который управляет вашей информацией для вас, помогает вам обнаруживать новую информацию оперативным путем, помогает вам получать и совместно использовать информацию с вашими коллегами и т.д.

Для достижения желательного уровня знания взаимодействие требует агентов, работающих в фоновом режиме, обосновывающих, изучающих, делающих логические выводы, согласовывающих пользователей на основе их профилей, получающих новые знания, автоматически логически выводящих новые знания и получающих знания из внешних источников таким образом, что они становятся неразрывной частью накопленного знания. Это в свою очередь требует семантической интеграции запасов знаний так, чтобы они имели смысл в целостной манере, а не просто обеспечивали базу для интеграции на уровне представления и поиска документов. Основная структура реализации и получаемая в результате среда должны обеспечивать услуги реального времени, обнаружения с высоким быстродействием и выдачи рекомендаций, так чтобы зависящая от контекста и времени информация признавалась особенной ценной, и чтобы специалисты в области информационных технологий работали более производительнее и получали больше результатов, быстрее и с меньшими потерями. И наконец, система должна работать с существующими источниками информации методом «включай и работай» (в режиме динамической настройки конфигурации), должна непрерывно и автоматически классифицировать известные запасы знаний и должна использовать инструментальные средства извлечения знаний в собственно знаниях, тем самым придавая еще одно «измерение» запасам знаний.

Настоящее изобретение направлено на интеллектуальную, проактивную, платформу знаний реального времени, которая сосуществует с современной сетью Web (или любым другим уровнем представления). Внедрение и использование настоящего изобретения позволит специалистам в области информационных технологий управлять их накопленными знаниями, поскольку авторская разработка (посредством «связей») будет осуществляться интеллектуально, динамически, автоматически и со скоростью мысли.

3. Современная «информационная» сеть Web в сравнении с информационной «нервной» системой согласно настоящему изобретению.

В среде современной сети Web семантика представляемой информации теряется после преобразования структурированных данных в HTML на сервере, что означает, что «знание» отделяется от объектов, прежде чем пользователь будет иметь возможность взаимодействовать с ними. Современная сеть Web создается авторами и «постоянно кодируется» на сервере, базируясь на представлении автора о том, как информация будет просматриваться и потребляться. Пользователи потребляют только информацию, как она представлена им.

Настоящее изобретение добавляет уровень интеллектуальности и уровень настройки, которые не могут поддерживаться в среде современной сети Web. Настоящее изобретение обеспечивает на XML-динамической сети Web объекты интеллектуального знания, вместо «тупых» Web-страниц, причем семантика объектов сохраняется между сервером и клиентом, обеспечивая пользователям больше возможностей управления накоплением их знаний. Кроме того, с использованием сети Web согласно изобретению специалисты в области информационных технологий могут потреблять информацию и воздействовать на нее, исходя из собственных понятий, поскольку они будут интерактивно создавать свои собственные накапливаемые знания посредством «динамического связывания» и «управляемого пользователем просмотра ресурсов».

Информационный агент (семантический браузер) в настоящем изобретении спроектирован для существования с современной сетью Web и интеграции и усиления всех граней частных и публичных интранетов и Интернет. Стеки технологических платформ современной сети Web и информационной нервной системы согласно настоящему изобретению приведены на фиг. 6. Как показано на фиг. 6, стек современной сети Web имеет самый нижний уровень источников структурированной информации, включая такую информацию, как данные, сохраненные в базах данных, и источников неструктурированной информации, включающих такую информацию, как документы, сообщения электронной почты и т.д. Информация на обоих этих уровнях обрабатывается отдельным образом. Никакая семантика не используется на уровне индексации информации; вместо этого используются поисковые процессоры, основанные на ключевых словах. Логический уровень состоит главным образом из базы данных, которая обеспечивает возможность программирования для поиска, правил, просмотра, запуска и т.д. Уровень приложений состоит из сценариев (скриптов) серверной стороны и приложений управления электронной коммерцией, основанных на пользовательском вводе. На самом верхнем уровне представления современная сеть Web имеет информацию представления (в форме Web-страниц), которая представляется посредством порталов на Web-платформе (например, браузеров).

Помимо перекрывающихся уровней обработки, настоящее изобретение уникальным образом обрабатывает информацию с самого нижнего уровня обработки таким образом, чтобы сохранить семантику источников информации. На обоих уровнях источников структурированной и неструктурированной информации система 10 обрабатывает информацию регулярным образом, принимая во внимание метаданные и семантику, связанные с информацией. На уровне индексирования информации метаданные информации и семантика извлекаются из неструктурированных данных. Система 10 добавляет три дополнительных уровня платформы, отсутствующих в современной сети Web: уровень индексации и классификации знаний, где информация структурированных и неструктурированных источников семантически кодируется; уровень представления знаний, где создаются связи, позволяющие самокоррекцию и исправление семантической сети объектов знаний; и уровень онтологии и логического вывода знаний, где логически выводятся новые соединения и свойства в семантической сети. На логическом уровне создается база знаний, которая обеспечивает программирование на семантическом уровне. На уровне приложений сценарии серверной стороны используются во взаимосвязи с базой знаний. Эти сценарии динамически генерируют объекты знаний на основе пользовательского ввода, и могут включать семантические команды для извлечения, извещения и логики. Этот уровень может также включать в себя интеллектуальных агентов для оптимизации обработки семантических пользовательских вводов. Уровень представления системы 10 сохраняет семантику, которая отслеживается с самых нижних уровней. Представление на этом уровне динамически генерируется на клиентской компьютерной системе и полностью настраивается.

За счет поддержки, интеграции и использования семантики на всех технологических уровнях настоящее изобретение создает виртуальную сеть Web дееспособных «объектов», которые непосредственно соответствуют «вещам», с которыми человек взаимодействует физически или виртуально или иными словами, известным «шаблонам контекста». В противоположность современной сети Web, которая является «тупой» сетью Web документов, настоящее изобретение обеспечивает интеллектуальную виртуальную сеть Web дееспособных объектов, которые имеют свойства и соотношения и в которых события могут вызывать изменения в других частях виртуальной сети Web.

Настоящее изобретение обеспечивает программируемую сеть Web. В отличие от современной сети Web, которая является «тупой» сетью Web документов, сеть Web согласно настоящему изобретению является программируемой, близкой базе данных, она способна обрабатывать логику и правила и будет иметь возможность инициировать события.

В то время как современная сеть Web кодируется для человека и поэтому главным образом фокусируется на представлении статической информации, виртуальная сеть Web согласно настоящему изобретению кодируется главным образом для машин, хотя в конечном счете обеспечивает представление для человека в качестве завершения цепочки доставки знаний. Настоящее изобретение обеспечивает интеллектуальную, обучающуюся сеть Web. Это означает, что виртуальная сеть Web в настоящем изобретении будет способна изучать новые соединения и становиться более интеллектуальной с течением времени. Сеть Web является динамической, виртуальной и самосоздаваемой, обеспечивая тем самым намного больше возможностей для специалистов в области информационных технологий за счет интеллектуальных и проактивно действующих семантических соединений, которые современная сеть Web не способна обеспечить, и приводя в результате к сокращению и в конечном счете к исключению потерь информации.

Сеть Web согласно настоящему изобретению является самокорректирующейся сетью. В отличие от современной сети Web, которая должна вручную поддерживаться создателями документов, настоящее изобретение обеспечивает сеть Web, которая самостоятельно поддерживается машинами. Это свойство восстанавливает разорванные связи, поскольку сеть Web будет автоматически фиксировать разъединения в сети.

Наконец, как будет более подробно изложено ниже, различные варианты осуществления настоящего

го изобретения воплощают некоторые или все из направлений приобретения знаний, описанных выше, для обеспечения существенных преимуществ над существующими системами, направленными на современную сеть Web или концептуальную семантическую сеть Web.

С. Архитектура системы и технологические факторы.

1. Обзор системы.

Настоящее изобретение направлено на систему и способ извлечения, управления и доставки знаний. Эта система и способ упоминаются здесь под зарегистрированным именем, выраженным термином Information Nervous System™ (информационная нервная система). Как показано на фиг. 7, на своем высшем уровне система 10 включает в себя сервер 20, состоящий из различных компонентов, которые работают вместе для обеспечения услуги извлечения зависимой от контекста и от времени семантической информации для клиентов 30, управляющих платформой представления (например, браузером) через коммуникационную среду 40, такую как Интернет или интранет. Серверные компоненты предпочтительно включают в себя сервер интеграции знаний (СИЗ) 50 и сервер базы знаний (СБЗ) 80, которые могут быть физически объединены или разделены. В системе все объекты или события в заданной иерархии являются активными агентами 90, семантически связанными друг с другом и представляющими запросы (составленные из кода основополагающих действий), которые возвращают объекты данных для представления клиенту в соответствии с предварительно определенной и настраиваемой темой (предметом) или «поверхностью» (оболочкой). Система предусматривает широкое разнообразие приложений, а также различные средства для клиента, чтобы настраивать и «сопрягать» агентов и связанные запросы, чтобы оптимизировать представление результирующей информации. Каждый из предпочтительных компонентов системы 10 настоящего изобретения, а также взаимодействие компонентов описаны более детально ниже.

2. Архитектура системы.

Сквозная системная архитектура для информационной нервной системы настоящего изобретения показана на фиг. 8. Фиг. 8 иллюстрирует, каким образом настоящее изобретение обеспечивает средство доступа множеству клиентов к информационному обмену между XML-Web-сервисом и интеллектуальными агентами информационной нервной системы. В предпочтительном варианте осуществления это происходит через информационного агента. В альтернативном варианте осуществления коммуникация может происходить программным образом через портал знаний предприятия (например, браузер доступа современной сети Web) или через уровень SDK, который обеспечивает программируемую интеграцию с настраиваемым клиентом.

Системная архитектура для СИЗ информационной нервной системы, включая ее компоненты, показана на фиг. 9. Эти компоненты описаны ниже более подробно.

3. Технологические наборы.

Существенные различия между современной сетью Web и концептуальной семантической сетью Web дополнительно пояснены со ссылками на технологические наборы, каждый из которых представлен на фиг. 10. Фиг. 10 показывает в высокоуровневом сопоставительном описании уровни платформы современной сети Web и соответствующие эквиваленты (если применимы) в информационной нервной системе настоящего изобретения. Фиг. 10 иллюстрирует, каким образом сценарии в современной сети Web отображаются на сценарии в информационной нервной системе в нескольких экземплярах, тем самым обеспечивая пользователей логическим маршрутом миграции, а также поясняет аспекты информационной нервной системы, которые не существуют в современной сети Web.

4. Гетерогенность системы.

Гетерогенность является преимуществом настоящего изобретения. В предпочтительном варианте XML-Web-сервис агентства СИЗ является переносимым. Это означает, что он поддерживает открытые стандарты, такие как XML, XML-Web-сервисы, которые являются совместимыми (например, используют стандарт WS-I взаимодействия), стандарты хранения и доступа к данным (например, SQL и ODBC/JDBC), стандартные протоколы для хранилищ информации, из которых DSA (агенты системы каталогов) собирают данные (например, LDAP, SMTP, HTTP и т.д.).

Например, в предпочтительном варианте осуществления СИЗ (на котором исполняется агентство) имеет возможность

собирать свои метаданные «люди» из памяти LDAP (с использованием LDAP DSA). Это позволяет поддерживать активный каталог Windows 2000 компании Microsoft, сервер каталога компании Sun и другие продукты каталогов, которые поддерживают LDAP. Предпочтительно иметь специфический для платформы DSA активный каталог, который использует специфические для платформы интерфейсы программирования приложений (API) для сбора метаданных «люди»;

собирать свои метаданные электронной почты из памяти SMTP (для электронной почты от любого источника или от системного почтового ящика). Это позволяет поддерживать Microsoft Exchange, Lotus Notes и другие серверы электронной почты (которые поддерживают SMTP). Предпочтительно иметь специфический для платформы DSA Microsoft Exchange Email или DSA Lotus Notes Email;

собирать свои метаданные «событие» из памяти календаря, поддерживающей открытый стандарт, подобный стандарту iCalendar и использовать протокол, такой как CAP (протокол доступа к календарю).

Это позволяет поддерживать любое хранилище событий, которое поддерживает стандарт iCalendar или стандарт протокола CAL. Предпочтительно иметь специфический для платформы DSA Microsoft Exchange Calendar (или Event) или DSA Lotus Notes Calendar и т.д.

В альтернативном варианте агентство СИЗ может конфигурироваться для выделения метаданных, сохраненных в частном (специализированном) хранилище (с соответствующим DSA).

Для достижения гетерогенности в предпочтительном варианте осуществления для коммуникаций клиент-сервер система 10 использует стандарты XML-Web-сервисов, которые работают во взаимодействии (между платформами). Это включает соответствующие открытые стандарты и стандарты взаимодействия для SOAP, XML, WS-Security, WS-Caching и т.д.

В предпочтительном варианте осуществления настоящего изобретения семантический браузер (также упоминаемый по его зарегистрированному имени Information Agent™ (информационный агент)) способен действовать между платформами и в различных средах, таких как Windows, NET J2EE, Unix и т.д. Эта способность согласована с замечанием о семантическом опыте пользователя, что пользователь не должен заботиться о том, на какой платформе исполняется браузер или на какой платформе исполняется агентство (сервер). Семантический браузер настоящего изобретения обеспечивает пользователей согласованным опытом независимо от того, общаются ли они с сервером Windows (или NET) или с сервером J2EE. От пользователей не требуется предпринимать особые шаги при установке или использовании клиента, основанного на платформе, на которой исполняется любое из агентств, с которыми они взаимодействуют.

Информационный агент предпочтительно использует открытые стандарты для своих поверхностей и других эффектов представления. Они включают в себя стандарты такие, как XSLT, SVG и специализированные форматы представления, которые работают на платформах (например, соответствующие версии Flash MX/ActionScript).

Простая гетерогенная сквозная реализация предпочтительного варианта осуществления информационной нервной системы настоящего изобретения показана на фиг. 11. Фиг. 11 иллюстрирует предпочтительный вариант осуществления информационной нервной системы и представляет гетерогенный «межплатформенный» контекст для настоящего изобретения. Компоненты, показанные на фиг. 11, более детально описаны ниже.

5. Защищенность.

Предпочтительный вариант осуществления информационной нервной системы обеспечивает поддержку для всех аспектов защищенности: аутентификации, авторизации, аудита, неприкосновенности данных, целостности данных, доступности, неизменности переданной информации. Это выполняется путем использования стандартов, таких как WS-Security, который обеспечивает платформу для защищенности для приложений XML-Web-сервиса. Защищенность предпочтительно обеспечивается на протокольном уровне за счет стандартов защищенности в стеке протоколов XML-Web-сервиса. Это включает в себя вызовы методов шифрования от клиентов (семантических браузеров) в серверы (агентства), поддержку цифровых подписей, аутентификацию вызывающего пользователя перед обеспечением доступа к семантической сети агентства и методы XML-Web-сервиса и т.д.

Предпочтительный вариант осуществления настоящего изобретения поддерживает локальное (клиентское) управление мандатом доступа. Это предпочтительно реализуется требованием от пользователей ввести список их пользовательских имен и паролей, которые они используют на множестве агентств (в интранет) или в Интернет. Семантический браузер агрегирует информацию от множества агентств, которые могут иметь различные аутентификационные мандаты для пользователя. Поддерживаемые аутентификационные мандаты дополнительно включают в себя общие схемы, такие как базовая аутентификация с использованием имени пользователя и пароля, базовой аутентификации в SSL, услугу аутентификации NET Passport (сетевой паспорт) компании Microsoft, новую услугу аутентификации Liberty Alliance (свободный альянс), сертификаты клиентов в SSL, цифровую аутентификацию и интегрированную идентификацию Microsoft (для использования в средах Microsoft).

В предпочтительном варианте, в котором пользовательские мандаты кэшированы у клиента, семантический браузер использует соответствующие мандаты доступа для конкретного агентства путем проверки поддерживаемого уровня и схемы аутентификации для данного агентства (который является частью схемы агентства). Например, если агентство поддерживает интегрированную Windows-аутентификацию, семантический браузер вызывает метод XML-Web-сервиса с обработкой входа в систему или другого идентификатора для текущего пользователя. Если агентство поддерживает только базовую аутентификацию в SSL, семантический браузер пропускает либо имя пользователя и пароль или кэшированную копию абстрактного идентификатора входа в систему (если клиент ранее регистрировался и идентификатор входа в систему не истек), чтобы осуществить регистрацию. Предпочтительный вариант осуществления использует методы, такие как кэширование идентификатора входа в систему, старение и истечение срока действия в СИЗ, чтобы ускорить процесс аутентификации (и поиски идентификаторов входа в систему) и обеспечить большую защищенность предохранением от пиратского использования идентификаторов входа в систему (регистрации).

XML-Web-сервис агентства предпочтительно поддерживает различные схемы аутентификации ли-

бо в неявном виде (если данная функция явно поддерживается операционной системой сервера или сервера приложений), либо на уровне приложений самой реализацией XML-Web-сервиса. Альтернативные варианты осуществления XML-Web-сервиса агентства СИЗ предпочтительно используют множество схем аутентификации, в том числе базовой аутентификации, базовой аутентификации в SSL, цифровой, интегрированной идентификации Windows, сертификаты клиентов в SSL и интегрированную услугу аутентификации NET Passport.

6. Факторы эффективности.

Кэши клиентских и серверных запросов и объектов.

Настоящее изобретение обеспечивает кэши запросов, которые ответственны за кэширование запросов для быстрого доступа. У клиента, кэш клиентских запросов кэширует результаты SQL-запросов с определенными аргументами. Кэш предпочтительно конфигурирован для очистки его содержимого спустя предварительно определенный интервал времени (например, несколько минут). Интервал времени предпочтительно устанавливается путем моделирования использования системы и получения оптимального значения для предела времени кэша. Другие параметры также могут учитываться, такие как скорость поступления данных в агентство (в случае кэшей на каждое агентство, что является другим вариантом реализации), модель использования (например, скорость навигации) пользователя и т.д.

Кэширование улучшает рабочие характеристики, так как клиент не должен без нужды получать доступ к недавно использовавшимся серверам по мере того, как он перемещается в семантической среде. В предпочтительном варианте клиент использует стандартные технологии кэширования XML-Web-сервиса (например, WS-кэширование). Кроме того, у клиента предпочтительно имеется кэш объектов. Этот кэш кэширует результаты каждого SQL-ресурса и маркируется ссылкой на ресурс (например, путь к файлу, URL и т.д.). Это оптимизирует SQL-обработку, поскольку клиент может получить XML-метаданные для SQL-ресурса непосредственно из кэша объектов, не получая доступ к собственно ресурсам. Ресурс может представлять собой локальную файловую систему, локальное приложение (например, Microsoft Outlook) или XML-Web-сервис агентства. Подобно кэшу запросов, кэш объектов может быть конфигурирован для очистки его содержимого спустя установленный интервал времени (например, несколько минут).

В альтернативном варианте на сервере серверный кэш запросов кэширует результаты категорий для XML-аргументов. Это ускоряет время отклика на запросы, поскольку серверу не требуется запрашивать KDM о категоризации XML-аргументов (через один или более экземпляров СБЗ, которые СИЗ конфигурировал для получения от него знания данной предметной области) по каждому запросу очереди. Кроме того, сервер может кэшировать SQL-эквиваленты SQL-аргументов, которые он получает от клиентов. Это ускоряет время реакции на запросы, поскольку серверу не нужно преобразовывать SQL-аргументы в SQL каждый раз, когда он получает запрос от клиента. В предпочтительном варианте используется агрессивное клиентское кэширование, и серверного кэширования можно избежать, если только оно явным образом не улучшает рабочие характеристики. Это объясняется тем, что клиентское кэширование масштабируется лучше, чем серверное кэширование, поскольку клиент кэширует запросы на основе его локального контекста.

Виртуальные, распределенные запросы.

Настоящее изобретение использует виртуальные, распределенные запросы. Это согласовано с его функциональными возможностями «динамического связывания» и «управляемого пользователем просмотра ресурсов». Система не требует статических сетей, которые связывают - или массивных индивидуальных баз данных, которые содержат - все метаданные для системы. Это исключает необходимость в ручном создании (авторской разработке) и поддержке в локальном или глобальном объеме. Кроме того, это исключает необходимость в интегрированном (или универсальном) хранилище, где должны храниться все требуемые метаданные, и которое доступно только через интерфейс запросов в базу данных (например, SQL). Вместо этого, настоящее изобретение использует принцип «динамического доступа» посредством использования XML-Web-сервисов для динамического распределения запросов по разным агентствам (контекстно- и времязависимым способом) и агрегирования результатов этих запросов согласованным и удобным для пользователя способом у клиента.

D. Компоненты и функционирование системы.

1. Агентства и агенты.

Настоящее изобретение вводит уникальный метод использования агентств и агентов для извлечения, управления и доставки знаний.

а. Агентства.

В предпочтительном варианте осуществления агентство - это экземпляр сервера интеграции знаний (СИЗ, KIS) 50, представляющий собой эквивалент в изобретении Web-сайта. Агентство предпочтительно инсталлировано как Web-приложение (на Web-сервере) для предложения XML-Web-сервисов. Агентство предпочтительно включает в себя администратора агентства. В предпочтительном варианте осуществления настоящего изобретения агентство имеет следующие основные компоненты:

флаг, указывающий, поддерживает ли агентство или требует аутентификации (или того и другого). Если агентство требует аутентификации, то агентство должно требовать базовую пользовательскую ин-

формацию и пароль и должно сохранять информацию о типе аутентификации, которую оно поддерживает. Для агентств, которые сохраняют пользовательскую информацию, агентство также будет требовать информацию о подписке пользователя (для подписки на агентов на конкретном агентстве);

структурированные хранилища семантических объектов (документов, сообщений электронной почты и т.д.) соответственно схемам для соответствующих классов;

компоненты времени исполнения, которые реагируют на семантические запросы, - компоненты возвращают XML на вызов приложения и обеспечивают системные сервисы для всех функций извлечения информации семантического браузера.

Серверное пользовательское состояние.

В предпочтительном варианте осуществления настоящего изобретения агентства поддерживают серверное пользовательское состояние, которое ассоциирует связанные концепты (элементы представления данных), включая метаданные «люди», и пользовательскую аутентификацию. Серверное пользовательское состояние облегчает множество деталей реализации настоящего изобретения, включая хранение любимых объектов пользователя (посредством семантических связей между объектами людей и информационными объектами), логический вывод любимых объектов (предпочтений), чтобы генерировать новые связи (например, рекомендации), аннотации (которые отображают пользовательские комментарии на информационные объекты) и логический вывод «экспертов» на основе семантических связей, которые отображают пользователей на информацию (например, посланные сообщения электронной почты, аннотации и т.д.). Серверное пользовательское состояние предпочтительно используется с некоторыми шаблонами контекста, такими как «эксперты», «предпочтения», «рекомендации» и «нюсмейкеры».

Клиентское пользовательское состояние.

Информационный агент (семантический браузер) предпочтительно поддерживает роуминг локального клиентского пользовательского состояния. Это включает в себя семантическую среду пользователя и мандаты доступа пользователя (переносимые защищенным образом). В предпочтительном варианте осуществления пользователи имеют возможность легко экспортировать свое клиентское пользовательское состояние на другую машину, чтобы дублировать свою семантическую среду на другой машине. Это предпочтительно достигается переносом пользовательского перечня агентов (последних и предпочтительных), метаданных для агентов (включая SQML-буферы), локально защищенные мандаты доступа пользователя и т.д. в XML-формат, который преобразует в последовательную форму все это состояние и обеспечивает возможность простого переноса состояния. Альтернативно, XML-схема может быть распространена на все локальное клиентское пользовательское состояние. Кэширование пользовательского состояния на сервере и синхронизация пользовательского состояния с использованием обычных методов синхронизации может также облегчить роуминг. Семантический браузер предпочтительно загружает и выгружает все клиентское пользовательское состояние на сервер вместо хранения состояния локальным образом (в XML-файле или специализированной памяти, подобной Windows-блоку регистров).

b. Агенты.

Агент есть главная точка входа в семантическую среду настоящего изобретения. Агент предпочтительно состоит из запроса семантического фильтра, который возвращает XML-информацию для конкретного типа семантического объекта (например, документы, электронная почта, люди и т.д.). Иными словами, агент предпочтительно конфигурируется с конкретным типом объекта (описано ниже). Агенты также могут конфигурироваться с шаблоном контекста (описано ниже). В этом случае запрос будет возвращать тип объекта, но он будет воплощать семантику шаблона объекта. Например, агенты, конфигурированные с шаблоном контекста «заголовки», будут сортироваться по времени и релевантности и т.д. Агенты также используются для фильтрации оповещений, предупреждений и извещений. Агентам может быть дано любое имя. Однако в предпочтительном варианте осуществления изобретения формат именования агентов следующий:

<Agentobjecttype>.<semanticqualifier>.<semanticqualifier>

Агенты могут быть поименованы произвольным образом. Однако примеры имен агентов включают следующее:

All.All (Все.Все)

Email.All (Электронная почта. Все) Documents.Technology.Wireless.80211B.All (Документы. Технология. Беспроводные системы.80211.B.Все)

Events.Upcoming.NextThirtyDays.All (События. Наступающие. Следующие три дня. Все)

Имеются также агенты предметной области (см. ниже), которые могут следовать отличающемуся соглашению о присвоении имен (см. ниже). В семантическом браузере настоящего изобретения полностью определенное имя агента предметной области может иметь следующий формат:

<Agentobjecttype>.<semanticdomainname>.<categoryname> [Agency=<Agency url>, kb=<kb url>]

Например, агент предметной области электронной почты в агентстве: <http://research.Agency.asp>, конфигурированный в категории wireless.all из базы знаний ABC.com/kb.asp с именем семантической предметной области industries.informationtechnology будет полностью поименован следующим образом:

Email.Industries.InformationTechnology.Wireless.All

[Agency=<http://research/Agency.asp>, kb=<http://abccorp.com/kb.asp>]

Семантический браузер настоящего изобретения предпочтительно конфигурируем для использования только имени агента или для включения спецификаторов «агентство» и «kb».

Типы агентов.

Имеется три основных типа агентов, создаваемых на сервере 20: стандартные агенты, составные агенты, агенты предметной области. Стандартный агент является автономным агентом, который инкапсулирует структурированные несемантические запросы, т.е. без знания предметной области (или отображение онтологии/таксономии). Например, на сервере агент All.PostedToday.All (Все.ПосланноеСегодня.Все) является простым агентом, который разлагается фильтрацией всех объектов на основе свойства время создания. Стандартные агенты также могут быть более сложными. Например, агент All.PostedByAnyMemberOf MyTeam.All (Все.ПосланноеЛюбымЧленомМоейКоманды.Все) может разлагаться в сложный запрос, который связан с присоединениями и подвопросами из таблицы объектов и таблицы пользователей (см. ниже).

Составной агент содержит других агентов и позволяет администратору агентства создавать запросы, которые генерируют результаты, которые представляют собой ОБЪЕДИНЕНИЕ или ПЕРЕСЕЧЕНИЕ результатов других содержащихся в нем агентов (в зависимости от конфигурации). Составные агенты могут также содержать других составных агентов. В предпочтительном варианте осуществления составные агенты содержат агентов из того же агентства. Однако настоящее изобретение допускает интеграцию агентов из других агентств. Например, составной агент All.Technology.Wireless.All может быть создан из следующих агентов: Documents.Technology.Wireless.All (Документы.Технология.Беспроводная.Все) Email.Technology.Wireless.All (ЭлектроннаяПочта.Технология.БеспроводнаяВсе) People.Experts.Technology.Wireless.All (Люди.Эксперты.Технология.Беспроводная.Все).

Как описано выше, агент предметной области является агентом, который принадлежит семантической области. Агент предметной области инициализируется запросом агента, аналогично любому другому агенту. Однако этот запрос включает в себя таблицу КАТЕГОРИИ, которая заполняется администратором предметной области знаний (см. ниже). Хотя предпочтительный вариант осуществления настоящего изобретения использует СБЗ 80, имеющий специализированные онтологии, соответствующие частной семантической среде, настоящее изобретение допускает поддержку стандартов взаимозаменяемости онтологии, которые позволят агенту соединиться с одной или несколькими настроенными частными СБЗ, например, с организацией, где агентство было первоначально инициализировано, со специализированной онтологией для данной организации.

Примером агента предметной области является следующий:

Email.Technology.Wireless.All. Этот агент предпочтительно создается с URL источника знаний, таким как category://technology.wireless.all@ABC.com/marketingknowledge.a sp

Этот URL источника знаний соответствует категории Technology.Wireless.All. для установленной по умолчанию области в базе знаний на Web-сервисе ABC.com/marketingknowledge.asp. Это разрешается на соответствующий HTTP URL: http://ABC.com/marketingknowledge.asp?category="technology.wireless.all."

В этом примере полностью определенная версия URL категории может быть следующей: category://technology.wireless.all@abccorp.com/marketingknowledge.asp?semanticdomain name="Information Technology".

В этом случае URL категории квалифицируется с именами предметных областей.

Агенты предметных областей предпочтительно создаются посредством мастера агента предметной области, и администратор агентства может дополнить агентов предметных областей из СБЗ 80 в семантическую сеть настоящего изобретения. Мастер агента предметной области позволяет пользователям создавать агентов предметных областей для конкретных категорий (используя URL категории) или для имени полной семантической области. В последнем случае агентство предпочтительно конфигурируется для автоматического создания агентов предметных областей, когда новые категории добавляются к семантической области на СБЗ. Эта функция позволяет областям и категориям оставаться динамическими и поэтому легко адаптируемыми во времени к потребности пользователя. Когда агенты предметных областей конфигурируются подобным образом, агентство является конфигурируемым для удаления агентов, которые больше не находятся в семантической области. По существу, в этом режиме агенты предметных областей синхронизированы с таблицей КАТЕГОРИЙ (которая в свою очередь синхронизирована с перечнем КАТЕГОРИЙ на релевантном СБЗ администратором предметной области знаний, как описано ниже).

Агент предметной области инициализируется структурированным запросом, который фильтрует данные, которыми агент управляет на основе имени категории или URL. Пример полученного в результате запроса для агента категории имеет вид

```
SELECT OBJECT FROM OBJECTS WHERE OBJECTID IN (SELECT OBJECTID FROM
SEMANTICLINKS WHERE PREDICATYPEID=50 AND SUBJECTID=1000 AND OBJECTID IN
(SELECT OBJECTID FROM CATEGORIES WHERE URL LIKE
category://technology.wireless.all@ABC.com/kb.asp?domain="marketing"))
```

В этом примере предикатный тип ИД «принадлежит к категории» предполагается имеющим значе-

ние 50, а ИД объекта категории предполагается имеющим значение 1000. Этот запрос может быть переведен следующим образом.

«Выбрать все объекты в агентстве, которое принадлежит к категории, объект которой имеет значение ИД объекта 1000 и URL которой имеет вид

`category://technology.wireless.all@ABC.com/kb.asp?domain="marketing"»`

Это, в свою очередь, может быть переведено так «Выбрать все объекты в агентстве категории `category://technology.wireless.all@ABC.com/kb.asp?domain="marketing"»`

Мастер агента предметной области запрашивает пользователя, желает ли он поименовать агента на основе короткого имени категории или удобной версии полностью определенного имени категории. Примером последнего является следующее:

`Marketing.Technology.Wireless.All [@ABC].`

Полностью определенное наименование агента предметной области будет следующим:

`<objecttypename>.<semanticdomainname>.<categoryname>.all [@KB Name].`

В этом примере имя агента предметной области есть

`Email.Marking.Technology.Wireless.All [@ABC].`

Элементы сопряжения.

Элементы сопряжения являются персональными суперагентами пользователя. Пользователи могут создавать элемент сопряжения и добавлять и удалять агентов (в агентстве) к/от элемента сопряжения. Это аналогично тому, что пользователи имеют свои собственные «персональные агентства». Элементы сопряжения предпочтительно вызываются только на системного клиента, поскольку они включают в себя агентов из множества агентств. Клиент согласно настоящему изобретению агрегирует все объекты из агентов элемента сопряжения и представляет их соответствующим образом. Элементы сопряжения предпочтительно включают в себя все характеристики манипулирования другими типами агентов, например, операцию графического интерфейса «перетащить и оставить», интеллектуальную лупу (см. ниже). Элемент сопряжения может содержать любой тип агента (например, стандартных агентов, агентов поиска, специальных агентов и другие элементы сопряжения).

Настоящее изобретение предусматривает мастера элемента сопряжения, который является пользовательским интерфейсом, спроектированным для помощи пользователям в создании элементов сопряжения. На фиг. 12-14 показаны виды экранов, иллюстрирующие отдельные аспекты пользовательского интерфейса мастера элемента сопряжения согласно предпочтительному варианту осуществления изобретения. На фиг. 12 показан вид экрана информационного агента, показывающий древовидное представление примера семантической среды и мастера «добавить элемент сопряжения», который позволяет пользователю создать и управлять новым элементом сопряжения. На фиг. 13 показана вторая страница мастера «добавить элемент сопряжения», где пользователи вводят имя и описание элемента сопряжения и дополнительно выбирают информацию фильтров типа объекта. На фиг. 14 показана третья страница мастера «добавить элемент сопряжения», согласно предпочтительному варианту осуществления изобретения. В этом примере пользователи добавляют и удаляют агентов из семантической среды к/из элемента сопряжения. Когда открыта опция «добавить агента», отображается диалог «открыть агента», из которого пользователи могут добавлять нового агента, элемент сопряжения или агентство к новому элементу сопряжения.

Агенты новостей, приводящих к прерыванию.

Агент новостей, приводящих к прерыванию, является специально маркированным интеллектуальным агентом. В дополнение к опции критичности ко времени, определяемой администратором агентства, пользователь имеет опцию указания, какие агенты ссылаются на информацию, о которой он хочет получать уведомления. Любая отображаемая информация будет показывать уведомления, если имеются новости, приводящие к прерыванию, связанные с ней на агенте новостей, приводящих к прерыванию. Например, пользователь может создать агента в следующем виде: «Все документы, посланные Reuters сегодня» или «Все события, относящиеся к компьютерной технологии и произошедшие в Сиэтле за последние 24 ч» как агента новостей, приводящих к прерыванию. Эта функция действует индивидуальным путем, поскольку каждый агент новостей, приводящих к прерыванию, является персональным (характеристика «прерывания» (чрезвычайности) является субъективной и зависит от пользователя). Например, пользователь в Сиэтле желал бы быть уведомленным о событиях, произошедших в Сиэтле за последние 24 ч, о событиях на Западном Побережье на следующей неделе (за это время ему надо отыскать дешевый полет), о событиях в США в следующие 14 дней (предварительное уведомление для большинства авиаперевозчиков в США, что получить сведения о 5 трансконтинентальных перелетах по умеренной цене), о событиях в Европе в следующем месяце (например, по причине обеспечения некоторого времени для резервирования места в отеле), и события где-либо в мире в течение шести следующих месяцев.

В предпочтительном варианте осуществления настоящее изобретение автоматически проверяет семантическую среду на наличие в ней новостей, приводящих к прерыванию, путем опроса каждого агента новостей, приводящих к прерыванию или путем запроса шаблона контекста «Новостей, приводящих к прерыванию». Оно будет делать это для всех объектов, отображаемых в окне семантического браузера. Если агент новостей, приводящих к прерыванию, указывает на наличие таких новостей, поверхность

объекта информационный агент указывает на это путем мигания окна или путем показа пользовательского интерфейса, который в явном виде указывает, что имеется уведомление, относящееся к объекту. Если пользователь выполнит щелчок на пиктограмме новостей, приводящих к прерыванию, то будет отображена панель новостей, приводящих к прерыванию, или палитра контекста для шаблона контекста «Новостей, приводящих к прерыванию», позволяя пользователю просматривать новости, приводящие к прерыванию, выбирать агента новостей, приводящих к прерыванию (если их имеется несколько), выбирать предикаты и выбирать другие опции. Пример панели пользовательского интерфейса агента новостей, приводящих к прерыванию, показан на фиг. 15. Этот пользовательский интерфейс показывает всплывающее меню в контекстной панели результатов. Показанный образец отображает аналогичную контекстную панель, что и в случае всплывающего меню в контекстной панели результатов интеллектуальной лупы (объекта агента), за исключением того, что здесь агент является агентом новостей, приводящих к прерыванию.

Устанавливаемые по умолчанию агенты.

В альтернативном варианте осуществления каждое агентство показывает перечень устанавливаемых по умолчанию агентов. Эти агенты сходны с устанавливаемой по умолчанию страницей Web-сайта; авторы агентства определяют, каких агентов пользователям желательно всегда видеть. Альтернативно, у клиента устанавливаемые по умолчанию агенты могут вызываться щелчком по корневому каталогу среды информационного агента (предпочтительно соответствует «исходному (домашнему) агенту», эквивалентному, например, «домашней странице» браузера современной сети Web). Пользователи могут также конфигурировать объединенных устанавливаемых по умолчанию агентов.

Устанавливаемые по умолчанию специальные (или контекстные) агенты.

В предпочтительном варианте осуществления клиент или агентство поддерживают устанавливаемого по умолчанию специального или контекстного агента, который отображается на каждый шаблон контекста (см. ниже). Эти агенты предпочтительно используют соответствующий шаблон контекста без какого-либо фильтра. Например, устанавливаемый по умолчанию специальный агент, называемый «Сегодня», возвращает все элементы по всем агентствам в перечнях «последнее» и «предпочтения» (или на конфигурированном перечне агентств) то, что было прислано сегодня. Еще в одном примере, устанавливаемый по умолчанию специальный агент, называемый «разное», показывает случайный набор результатов для каждого агентства в семантической среде соответственно шаблону контекста «разное».

Устанавливаемые по умолчанию специальные агенты предпочтительно действуют как исходная точка для большинства пользователей, чтобы познакомиться с информационной нервной системой настоящего изобретения. Кроме того, устанавливаемые по умолчанию специальные агенты сохраняют ту же функциональность, что интеллектуальные агенты, включая операцию графического интерфейса «перетащить и оставить», копировать и вставить, интеллектуальную лупу, глубокую информацию и т.д.

Агенты горизонтального решения.

В предпочтительном варианте осуществления агенты, используемые клиентом для помощи во взаимодействии с пользователем, включают следующее:

Агент планирования - агент планирования интеллектуальным образом ранжирует события на основе вероятности того, что конкретные пользователи пожелают уделить внимание событию.

Агент повторной встречи - агент повторной встречи интеллектуальным образом уведомляет пользователей, когда подошло время, о повторной встрече с тем, что произошло в прошлом. Процессор логического вывода (см. ниже) контролирует соответствующую семантическую деятельность для определения того, произошло ли достаточно изменения в качестве основания для повторной встречи. Пользователи предпочтительно используют объект предшествующей встречи в качестве опорной точки информационного объекта, чтобы найти изменения релевантных знаний (такие как новые документы, новые люди, которых желательно посетить и т.д.).

Агент повторной задачи. Агент повторной задачи посылает рекомендации пользователям в ответ на задачи, выполняемые пользователями (такие как чтение документа, добавление события в календарь и т.д.). Агент гарантирует пользователю постоянное отслеживание. Рекомендации базируются на пользовательских профилях, и агент предпочтительно использует совместную фильтрацию для определения рекомендаций.

Агент повторной настройки. Агент повторной настройки посылает уведомления пользователям на основе их деятельности. Агент интеллектуальным образом определяет, когда пользователь требует внимания (на основе электронной почты, принятой пользователем, новых документов, которые могут помочь обслуживанию пользователя и т.д.).

Публичные и локальные агенты. Агенты, которые создаются администратором агентства, являются «Публичными (общедоступными) Агентами». Агенты, создаваемые и управляемые пользователями, являются «Локальными Агентами». Локальные агенты могут ссылаться на удаленные агентства через SQML, которые включают в себя ссылки на URL XML-Web-сервисов Агентства, или могут ссылаться на локальные агентства, которые исполняют локальные экземпляры СИЗ с локальной памятью метаданных.

Пользовательский перечень «Мои Агенты» - «Сохраненные Агенты». В предпочтительном варианте пользователи могут сохранять копию вызванного агента или результат запроса в качестве локального

агента. Например, пользователи могут «перетащить и оставить» документ на своем жестком диске в папку агента для генерации семантического реляционного запроса. Пользователи могут сохранить этот результат как агента с именем "Documents.Technology.Wireless.RelatedToMyDocument." Это позволит затем пользователю перемещаться к этому агенту для поиска персонализированного семантического запроса.

Пользователи будут иметь возможность использовать этого агента для создания новых персональных агентов и т.д. Персональные агенты могут затем «публиковаться» в агентстве. Другие пользователи предпочтительно могут обнаруживать агента и подписываться на него.

В предпочтительном варианте локальный агент создается кнопкой "Save as Agent" (Сохранить как Агента), которая представлена на клиенте всегда, когда отображается результат семантического реляционного запроса. Это аналогично сохранению пользователями нового документа. Как только Агент сохранен, он добавляется к пользовательскому перечню «Мои Агенты». Агент реагирует на семантический запрос, основанный на семантической области агентства, на котором он находится. По существу, семантический запрос к агенту аналогичен вопросу, понятен ли для агента запрос. Агент реагирует на запрос наилучшим образом из того, что ему понятно. В качестве дальнейшей иллюстрации агент, который управляет объектом «Люди», реагирует на семантический запрос запрашиванием у экспертов документа, основываясь на собственном внутреннем отображении людей в своей семантической области на категории в этой области.

Альтернативно, клиент системы может быть конфигурирован для использования несемантических запросов. В этом случае агентство будет использовать выделенные ключевые слова для запроса. Все агенты поддерживают несемантические запросы. Предпочтительно только агенты в агентстве, которое принадлежит к семантической области, будут поддерживать семантические запросы. Иными словами, семантические поиски сводятся к обычным поискам.

Каждый агент содержит атрибут, указывающий, является ли он «интеллектуальным» или нет. Интеллектуальный агент предпочтительно создается на агентстве, если это агентство принадлежит к семантической области. Кроме того, интеллектуальный агент только возвращает объекты, которые он полностью «понимает». В предпочтительном варианте, если агентство инсталлировано, то имеется множество установленных по умолчанию интеллектуальных агентов, которые администратор агентства может выбирать для инсталляции, включая следующие:

All.Understood.All
Documents.Understood.All
Email.Understood.All

Например, Email.Understood.All возвращает только объекты электронной почты, которые Агентство может семантически понимать на основе своей семантической области (или онтологии).

Настоящее изобретение предпочтительно включает в себя возможности для пользователя отображать все объекты, и только те, которые агентство понимает.

Агенты поиска. Агент поиска является агентом, который инициализируется строкой поиска. В предпочтительном варианте при инициировании работы клиент выдает запрос поиска. Агент поиска конфигурируется так, чтобы осуществлять поиск в любой части семантической среды, включая

часто используемые агенты;
недавно использованные агенты;
недавно созданные агенты;
предпочтительное;
все (сохраненные) агенты;
удаленные агенты;
агенты в локальной сети;
агенты в глобальном каталоге агентства;
агенты в настроенных пользователем каталогах агентства;
все агенты во всей семантической среде.

Клиент выдает запрос на поиск на основе объема агента поиска. Если пользователи указывают, что они хотят осуществить поиск, чтобы перекрыть всю семантическую среду, клиент выдает запрос ко всем агентам в администраторе семантической среды (см. ниже) и всем агентам в локальной сети, в глобальном каталоге агентства и в настроенных пользователем каталогах агентства.

Серверные предпочтительные агенты. Еще в одном альтернативном варианте осуществления агентство поддерживает предпочтительных агентов поддержки пользовательского состояния. В контексте, аналогичном современной сети Web, Web-сайт позволяет пользователям настраивать свои предпочтительные связи, документы и т.д. При первоначальном запросе агентство отображает как устанавливаемых по умолчанию агентов, так и предпочтительных агентов вызывающего пользователя (если имеет место пользовательское состояние).

Интеллектуальные агенты. Интеллектуальный агент является автономным агентом, который инкапсулирует структурированные семантические запросы, которые ссылаются на агентство через их XML-Web-сервис. В предпочтительном варианте осуществления пользователь клиента может создавать и редактировать интеллектуальных агентов посредством мастера «создать интеллектуального клиента», что

позволяет ему просматривать ресурсы семантической среды через диалог «открыть агента» и добавлять связи из определенных агентств. По существу, это соответствует созданию пользователями SQL-запроса из пользовательского интерфейса. В предпочтительном варианте осуществления пользовательский интерфейс только позволяет пользователям добавлять связи из ресурсов того же самого агентства. Однако пользователи могут создавать агентов одинаковых категорий среди агентств, в дополнение к специальным агентам и элементам сопряжения (которые являются предпочтительно перекрестными агентствами). Пользовательский интерфейс позволяет пользователю добавлять связи с использованием существующих интеллектуальных агентов в качестве опорных пунктов информационных объектов, при условии, что интеллектуальный агент обращается к тому же самому агентству для текущего запроса. Фиг. 16 иллюстрирует предпочтительный вариант, показывающий диалог «открыть агента» со средствами управления пользовательского интерфейса для выбора шаблонов связей (предикатов), самих связей и объектов. Фиг. 17-19 показывают древовидное представление примера семантической среды, связанной с диалогом «открыть агента». Фиг. 17 показывает диалог «открыть агента», позволяющий пользователям просматривать ресурсы семантической среды и открывать агента. Фиг. 18 иллюстрирует способ оперирования агентствами в семантической среде и диалог «открыть агента» с представлением «мелкий предварительный просмотр». Фиг. 19 иллюстрирует инструментальное средство «открыть» на панели инструментов, показывающее новые опции для открытия агентов из семантической среды или для импорта регулярной информации (например, из файловой системы) в семантическую среду путем создания неинтеллектуальных агентов.

Шаблоны связей, по существу, позволяют пользователю перемещаться по предикатам для текущего типа объекта с использованием предварительно определенных фильтров, тем самым позволяя пользователю избежать прохождения через все предикаты типа объекта. Примеры шаблонов связей:

Все

Новости, приводящие к прерыванию (связи, которые ссылаются на временную зависимость, например, «послано в прошлом»)

Категоризация

Определенный (невероятные связи)

Вероятный (вероятные связи)

Аннотации

В предпочтительном варианте осуществления диалог «открыть агента» позволяет пользователю выбрать объект для «связи с», и в зависимости от типа объекта позволяет просматривать объект (например, из средства управления календарем, если он представляет собой дату/время, из текстового окна, если он является текстом, из файловой системы, если он является файлом или путем каталога и т.д.). Пользовательский интерфейс мастера также позволяет пользователю предварительно просмотреть результаты запроса. Временная SQL-запись создается с текущим перечнем предикатов, и он загружается в окно мини-браузера в диалоговом окне мастера. Пользователь имеет возможность добавлять и удалять предикаты, а также будет иметь опцию указания того, желательна ли ему объединение («ИЛИ») или пересечение («И») предикатов. Пользовательский интерфейс также будет проверять наличие дублирования предикатов.

Как только пользователь завершает работу с мастером по созданию интеллектуального агента, интеллектуальный агент добавляется к семантической среде, и SQL также сохраняется вместе с ассоциированной записью объекта. В предпочтительном варианте пользователь может позже просмотреть интеллектуального агента с использованием таблицы свойств инспектора (программы контроля состояния сложных структур данных) свойств агента. Это позволяет пользователю просмотреть простые свойства семантической среды (например, имя, описание, время создания и т.д.), а также просмотреть URL ресурсов (WSDL URL для XML-Web-сервиса агентства, в которое направляется запрос) и перечень предикатов. Пользователь может редактировать перечень из таблицы свойств.

Устанавливаемый по умолчанию интеллектуальный агент. Устанавливаемый по умолчанию интеллектуальный агент подобен устанавливаемому по умолчанию специальному агенту за исключением того, что основывается на типах информационных объектов, а не на шаблонах контекста. Например, «документы» будут возвращать все документы по всем агентствам в пользовательской семантической среде; «электронная почта» будет возвращать все сообщения электронной почты в пользовательской семантической среде и т.д.

Специальный агент. Специальный агент является интеллектуальным агентом, созданным пользователями на основе шаблона контекста (см. ниже). Специальный агент предпочтительно инициализируется с именем агента, хотя и без ссылки на конкретного агента. Например, специальный агент "Email.Technology.Wireless.All" (Электронная почта.Технология.Беспроводная.Все) может быть создан, даже если нет агентов с таким именем в семантической среде. Подобно агенту поиска, специальный агент направлен на поиск любого агента с таким именем в любой части семантической среды. В предпочтительном варианте осуществления, когда специальный агент вызывается пользователями, клиент осуществляет поиск агентов, которые носят его имя. Если он находит каких-либо агентов, с этим именем, то клиент вызывает агента.

В предпочтительном варианте осуществления пользователи вводят параметры, согласованные с шаблоном контекста, указывая фильтры категории (если необходимы) и то, какие агентства запросить. Это может быть вручную введено с использованием диалога «открыть агента», или пользователи могут пожелать запросить «последних» агентов, «предпочтительных» агентов, или и тех и других. В альтернативном варианте осуществления пользователи могут выбрать категории (если требуется), которые объединены или пересекаются с выбранными агентствами, или все категории, известные в глобальном каталоге агентств. Еще в одном альтернативном варианте осуществления пользователи могут выбрать тип информации (в противоположность шаблону контекста) и ключевые слова для поиска (в противоположность предикатам или категориям).

Устанавливаемые по умолчанию специальные агенты. В предпочтительном варианте осуществления клиент системы устанавливает устанавливаемых по умолчанию специальных агентов, которые отображаются на все поддерживаемые шаблоны контекста. Например, в предпочтительном варианте осуществления устанавливаемые по умолчанию специальные агенты включают следующее:

- Заголовки
- Новости, приводящие к прерыванию
- Диалоги
- Ньюсмейкеры
- Наступающие события
- Обнаружение
- Архив
- Все варианты выбора
- Наилучшие варианты выбора
- Эксперты
- Предпочтения
- Классика
- Рекомендации
- Сегодня
- Разнообразие
- Временная ось
- Наступающие события
- Руководство

Заказные специальные агенты. В противоположность созданным пользователем специальными агентами, заказные специальные агенты являются специальными агентами, специально разработанными и удостоверяемыми, чтобы гарантировать, что специальные агенты являются надежными, защищенными и высокоэффективными. Настоящее изобретение обеспечивает уровень расширений, дающий возможность организациям и разработчикам создавать свои собственные заказные элементы сопряжения. Примером заказного элемента сопряжения может служить «Все.КритическийПриоритет. Все, что относится к моим самым последним документам или электронной почте». Этот заказной элемент сопряжения может быть реализован посредством SQML-файла с записью ресурса следующим образом:

```
<resource type= "nervana:url"
agent://all.criticalpriority.all@localhost>
<link predicate="nervana:relevantto" type= "nervana:localsemanticref recentdocuments>
<link operator= "or" type= "nervana:localsemanticref recentemail>
</resource>
```

В предпочтительном варианте осуществления презентатор (см. ниже) разрешает запись "link" («связь») локально и инициирует запросы XML-Web-сервиса к целевому ресурсу с XML-аргументами, соответствующими новейшим документам или сообщениям электронной почты.

Это позволяет целевому агенту фокусироваться на реагировании на семантические запросы исключительно XML-фильтрами, без знания семантики, связанной с происхождением фильтра. В альтернативном варианте заказной элемент сопряжения, как в приведенном выше примере, является устанавливаемым по умолчанию агентом.

Агенты вертикального решения. Агенты вертикального решения являются агентами, которые обеспечивают поддержку решения для вертикальных промышленных сценариев.

Схема агента. Агенты действуют в пределах определенных параметров и проявляют определенные характеристики, которые образуют схему агента. Схемы агентов могут варьироваться в широких пределах, будучи применимыми в рамках технологии настоящего изобретения. Для примера, схема агента согласно настоящему изобретению показана на фиг. 20. Настоящее изобретение особым образом предусматривает дополнение других полей. Например, поля URL (или пути) категории и имя шаблона контекста могут быть добавлены к схеме агента, чтобы обеспечить клиенту и серверу быстрый доступ к категории и шаблону контекста, который представляет агент (если применимо). Это полезно для администратора семантической среды, чтобы обеспечивать различные виды агента (по категории, по контексту и т.д.). Это дополняет существование этих полей в SQML для агента (выраженное через атрибуты и/или

предикаты).

Идентификаторы типов агентов (AgentTypeIDс), включенные в предпочтительный вариант осуществления, показаны на фиг. 21. Идентификаторы типов запросов агентов (AgentQueryTypeIDс), включенные в предпочтительный вариант осуществления, показаны на фиг. 22.

В предпочтительном варианте осуществления используются SQL-форматы запросов. Однако множество форматов запросов, например, XQL, XQuery и т.д. также включены в объем настоящего изобретения.

СИЗ 50 предпочтительно содержит таблицу агентов (для серверных агентов) в своей памяти данных, соответствующую этой схеме. Фиг. 23 иллюстрирует примеры семантических запросов, которые соответствуют именам агентов, показывая, как серверные агенты предпочтительно конфигурируются в СИЗ настоящего изобретения.

Как пояснено более детально ниже, агенты могут факультативно включать в себя свои собственные поверхности. Поверхность агента представлена как URL для XSLT-файла или эквивалентно Flash MX или ActionScript. Если URL поверхности агента не определен, то предполагается устанавливаемая по умолчанию поверхность для данного типа объекта агента.

Правила запроса агента. Каждый запрос серверного агента должен определяться для возврата столбца идентификатора объекта (OBJECTID). Каждая таблица имеет этот столбец, который является тем, что связывает таблицу объектов с таблицами выведенных типов объектов. Объекты и другие таблицы описаны более детально ниже.

Поскольку каждый запрос агента может формировать базе подзапроса, каскадированный запрос или объединение, предпочтительно, чтобы каждый запрос следовал этому формату. Например, запрос всех новостей "News.All" может быть таким: "SELECT OBJECTID FROM NEWS" (выбрать ИД объектов из новостей) (здесь "NEWS" есть имя таблицы, содержащей метаданные для новостных статей со схемой «новости»). В результате сервер 10 может затем использовать этот запрос как часть комплексного запроса. Например, если пользователь перемещает и отпускает документ в агента, сервер может выполнить этот запрос как

```
SELECT OBJECTID FROM NEWS WHERE OBJECTID IN (SELECT OBJECTID FROM
SEMANTICLINKS WHERE SUBJECTID IN (50, 67, 89) AND LINKSCORE > 90)
```

В данном примере предполагается, что документ классифицируется для отнесения к категориям в таблице КАТЕГОРИИ с идентификаторами 50, 67 и 89 объекта и что вероятность связи 0,9 есть порог для установления того, что документ принадлежит к категории. В этом примере документ используется как фильтр для запроса "News.All" и текст запроса используется как часть комплексного запроса.

Обладание согласованным стандартом для запросов позволяет процессору семантических запросов объединять запросы до тех пор, пока, наконец, не потребуются их представление. Например, каждое обращение к процессору семантического запроса должно указывать, какой тип объекта в каком должен возвращаться в результате. Затем процессор запросов возвращает XML информацию, согласованную со схемой для запрошенного типа объекта. Иными словами, процессор запросов предпочтительно возвращает специфические для схемы результаты для представления. Каждый запрос сохраняется на семантическом уровне (для возврата OBJECTID). Для использования последнего примера, когда пользователь вызывает агента News.All, браузер обращается к процессору запросов на XML-Web-сервисе агентства. Процессор запроса затем вызывает запрос и фильтрует его типом объекта "News Article" («новости статья»)

```
SELECT * FROM NEWS WHERE OBJECTID IN (SELECT OBJECTID FROM NEWS)
```

Это возвращает все поля для схемы «Новости». Браузер (через презентатора) отображает информацию с использованием XSLT (или инструментального средства представления, такого как Flash MX или ActionScript) либо для поверхности агента, либо для определенной пользователем поверхности (которая будет переопределять поверхность агента).

Параметры виртуального запроса. Запросы агента предпочтительно содержат специальный виртуальный параметр. Типовой пример может включать "%USERNAME%" («имя пользователя»). В этом примере процессор семантических запросов (SQP) преобразует виртуальный параметр в реальный аргумент перед вызовом запроса. Агент People.MyTeam.All («Люди.МояКоманда.Все») конфигурируется в SQL-запрос

```
SELECT * FROM USERS WHERE Division IN (SELECT Division FROM
USERS WHERE Name LIKE %USERNAME%)
```

В данном примере имя агента включает "MyTeam", даже если агент может быть применен к любому пользователю. Переменная %USERNAME% разрешается посредством SQP в имя реального вызывающего пользователя. SQL-вызов может быть разрешен в следующую форму:

```
SELECT * FROM USERS WHERE Division IN (SELECT Division FROM
USERS WHERE Name LIKE JohnDoe)
```

В этом примере JohnDoe предполагается именем вызывающего пользователя.

Простой поиск агента. Каждый агент поддерживает функциональные средства простого поиска. В предпочтительном варианте осуществления может выполнить щелчок правой кнопкой на интеллекту-

альном агенте в информационном агенте и выбрать «Поиск». Это вызовет диалоговое окно, в которое пользователь вводит текст для поиска. Это создает соответствующий SQML с соответствующим предикатом, например, "nervanaxontain".

Настоящее изобретение предусматривает простой, быстрый путь для пользователей, чтобы отыскивать агентов (и создавать оттуда интеллектуальных агентов) без прохождения через мастера «Создать интеллектуального агента» и выбора предиката «содержит текст» (что, альтернативно, достигает тот же результат).

Представления агентов агентства. Альтернативный вариант осуществления изобретения включает в себя представления агентов агентства. Представление агентов агентства является запросом, который фильтрует агентов на основе предварительно определенных критериев. Например, представление агента «документы» возвращает только агентов, которые управляют объектами семантического класса документов. Представление агента «Новости Reuters» возвращает перечень агентов, которые управляют объектами новостей с «Reuters» в качестве издателя. Представления агентов агентства являются важными, чтобы дать пользователям простой способ перемещения между агентами. Администратор агентства может создавать и удалять представления агентов.

Публикация и совместное использование агентов.

Предпочтительный вариант осуществления обеспечивает возможность простой публикации и совместного использования агентов. Это предпочтительно реализуется путем преобразования в последовательную форму семантической среды в XML-документ, содержащий последних и предпочтительных агентов, их схему, их SQML-буфера и т.д. и публикации документа в пункте публикации (доступном виртуальном каталоге). Этот XML-документ может также быть послан по электронной почте коллегам, друзьям и т.д., чтобы облегчить распространение и совместное использование локальных (созданных пользователями) агентов. Это аналогично тому, как Web-страницы публикуются в настоящее время и как Web URL и связи совместно используются путем посылки связей и приложений по электронной почте.

2. Сервер интеграции знаний.

Сервер интеграции знаний (СИЗ, KIS) 50 является центральным звеном серверной стороны системы 10. KIS семантически интегрирует данные от множества различных источников семантическую сеть и содержит агентов, которые обеспечивают доступ к сети. KIS также содержит семантические XML-Web-сервисы для обеспечения клиентам доступа к семантической сети через агентов. Для пользователей инсталляция KIS может представляться как агентство. KIS предпочтительно инициализируется со следующими свойствами:

Имя агентства. Имя агентства, например, «ABC»

Дружественное (сетевое) имя агентства. Полное имя агентства, например, «ABC Corporation»

Описание агентства. Описание агентства

Системное пользовательское имя агентства.

Пользовательское имя агентства. Каждое агентство представлено пользователем в каталоге предприятия (или Web-сайта), на котором оно установлено. Системное пользовательское имя используется для ведения системного входного почтового ящика (через который пользователи будут публиковать документы, электронную почту и аннотации для агентства). Для аутентификации агентство должно быть инсталлировано на сервере, который имеет доступ к учетным записям пользователя системы.

Уровень поддержки аутентификации агентства. Указывает, поддерживает ли агентство или требует аутентификации пользователя. Агентство может быть конфигурировано, чтобы не поддерживать аутентификацию (в этом случае оно открыто для всех пользователей и не имеет какого-либо пользовательского состояния), чтобы поддерживать, но не требовать аутентификацию, и чтобы требовать аутентификацию, и в этом случае оно предпочтительно указывает тип шифрования аутентификации.

Тип пользовательского каталога агентства. Это указывает тип пользовательского каталога, по отношению к которому агентство аутентифицирует пользователей и откуда агентство получает информацию о пользователях. Например, это может быть каталогом LDAP, пользовательским каталогом Microsoft Exchange 2000, пользовательским каталогом Lotus Notes на активном каталоге Windows 2000 и т.д.

Имя пользовательского каталога агентства. Это указывает имя сервера пользовательского каталога агентства (например, имя сервера Microsoft Exchange 2000).

Имя пользовательского домена агентства. Это указывает имя пользовательского домена для целей аутентификации. Это поле является факультативным и включается только в том случае, когда агентство поддерживает аутентификацию.

Имя группы пользователей агентства. Это указывает имя группы пользователей для целей аутентификации. Например, агентство может быть инициализировано с именем домена "USEmployees" и с именем группы "Marketing". В этом случае агентство будет сначала проверять имя пользователя, чтобы убедиться, что пользователь является членом данной группы пользователей, а затем направлять запрос аутентификации в аутентификатор пользовательского каталога, указанный типом пользовательского каталога. Если вызывающий пользователь не является членом группы пользователей, то запрос аутентификации отклоняется. Это поле действует, только если агентство поддерживает аутентификацию.

Имя соединения с информационным хранилищем. Это указывает имя соединения с хранилищем ба-

зы данных. Оно может быть представлено, как, например, имя ODBC-соединения в Windows (или как JDBC-имя и т.д.). KIS будет использовать базу данных, упоминаемую по имени соединения, для хранения, обновления и поддержания ее таблиц (см. ниже).

Динамическая оценка свойств. XML-Web-сервис агентства предпочитательно использует методы для возврата динамических свойств, например, список путей семантических предметных областей, которые сервер в текущий момент поддерживает или «понимает». Это позволяет пользователям перемещаться по агентству у клиента с использованием поддерживаемых путей семантических предметных областей или онтологий/таксономий. Как показано на фиг. 24, KIS 50 предпочитательно включает в себя следующие основные компоненты: семантическую сеть 52, компонент 54 сбора данных, компонент 56 контроля согласованности семантической сети, процессор 58 логического вывода, процессор 60 семантических запросов, компонент 62 синтаксического анализа естественного языка, агент 64 знаний электронной почты и администратор 66 предметных областей знаний.

а. Семантическая сеть.

Семантическая сеть есть ключевой информационный компонент сервера интеграции знаний (KIS). Семантическая сеть связывает объекты определенных схем настоящего изобретения вместе семантическим способом через таблицы баз данных. Семантическая сеть состоит из схем и хранилища семантических метаданных (SMS). Семантическая сеть предпочитательно состоит из двух схем данных: объекты и семантические связи. Дополнительные информационные схемы могут быть включены на основе системных требований и нужд предприятий. SMS предпочитательно представляет собой стандартную базу данных (SQL-сервер, Oracle, DB2 и т.д.), где все семантические данные хранятся и обновляются посредством таблиц баз данных. SMS предпочитательно включает в себя таблицы для каждого основного типа объекта (см. ниже).

К примеру, семантическая сеть, соответствующая ситуации предприятия, показана на фиг. 25, где иллюстрируются связи между бизнес-пользователями согласно настоящему изобретению и различными источниками и результатами извлечения, управления, доставки и представления знаний.

Объекты. Таблица «Объекты» содержит каждый объект в семантической сети. «Объект» можно представить как «базовый класс», из которого должен выводиться каждый тип семантического объекта. Предпочтительная схема типа объектов представлена на фиг. 26. ObjectID (идентификатор объекта) является уникальным идентификатором, который маркирует объект в семантической сети. Каждый объект в системе будет иметь схему, которая является расширением схемы объектов. Альтернативно, типы семантических объектов (например, документ, электронная почта, событие и т.д.) будет иметь только поле ObjectID. При направлении запроса процессор запросов может агрегировать информацию из таблицы объектов и конкретной семантической таблицы для получения окончательных результатов. Первый подход (с использованием каждой схемы, являющейся расширением схемы объекта) приводит в результате к лучшей характеристике времени исполнения, поскольку исключаются объединения. Однако последний подход, хотя и более дорогостоящий с точки зрения вычислений, приводит в результате к меньшим потерям в использовании памяти. ObjectTypeID (идентификатор типа объекта) предпочитательно является числом, которое преобразуется в последовательность, описывающую иерархию типа объекта, например

"documents\documents" (документы/документы);

"documents\analyst briefs" (документы/резюме специалистов); and "events\meetings" (события/встречи).

The SourceID (идентификатор источника) относится к идентификатору для адаптера семантических данных (SDA), от которого были получены данные объекта. Компонент сбора семантических данных (SDG) использует эту информацию для периодической проверки того, существует ли еще объект, путем запроса информации от SDA, из которого был извлечен объект.

Семантические связи. SMS предпочитательно включает схему «Семантические связи» (и соответствующую таблицу базы данных), которая сохраняет семантические связи. Эти связи будут аннотировать объекты в других таблицах данных SMS и предпочитательно образовывать модель данных для семантической сети. Каждая семантическая связь должна иметь ИД семантической связи. Таблица «Семантические связи» предпочитательно включает в себя имена и типы полей, как показано на фиг. 27. SubjectID и SubjectTypeID являются идентификатором объекта и идентификатором типа объекта для объекта, от которого исходит связь. ObjectID и ObjectTypeID являются идентификатором объекта и идентификатором типа объекта для объекта, к которому идет связь. LinkScore (ранг связи) предпочитательно ранжируется в пределах от 0 до 100 и представляет семантическую мощь связи как вероятности. Эти поля являются только иллюстративными; больше предикатов предусматривается на основе конкретного типа объекта, а также желания пользователя относительно семантических связей. Предпочтительный вариант осуществления изобретения предусматривает ИД типов предикатов, показанные на фиг. 28. Настоящее изобретение предусматривает дополнение других ИД типов предикатов.

Для примера, семантическая связь «Стив сообщает Патрику» будет представлена в таблице с SubjectID, соответствующим ИД Стива в таблице «Пользователи», типом предиката PREDICATETYPE_REPORTSTO («тип предиката - сообщает к») (см. таблицу ниже), ObjectID, соответствующим Патрику в таблице «Пользователи», рангом связи 100 (указывающим состояние как «истинно», то

есть связь не является вероятностной) и справочными данными, которые определяют связь. KIS создает, обновляет и поддерживает таблицы базы данных для каждого типа объекта (через SMS). Приведенное ниже иллюстрирует предпочтительный, но не исчерпывающий перечень основных и производных типов объектов:

- Персона
- Пользователь
- Клиент
- Категория
- Документ
- Резюме специалиста
- Отчет специалиста
- Анализ проблемы
- Белая книга (официальное издание)
- Профиль компании
- Электронная книга
- Электронный журнал
- Сообщение электронной почты
- Аннотация электронной почты
- Рассылка новостей по электронной почте
- Список распределения электронной почты
- Общедоступный каталог электронной почты
- Новостная группа каталога электронной почты
- Новостная статья
- Событие
- Встреча
- Корпоративное событие
- Отраслевое событие
- ТВ событие
- Радио событие
- Событие средства печати
- Онлайновая встреча
- Событие в мире искусства и развлечений
- Онлайновый курс обучения
- Печать
- Книга
- Журнал
- Мультимедиа
- Интерактивное вещание
- Интерактивная конференция

Типы объектов обычно выражаются как иерархические пути. Путь может быть расширен, например, «события/встречи» может быть расширено как «определенные встречи», например, «события/встречи/встречи в компании». Эта модель схемы является наращиваемой и конфигурируемой.

Типы виртуальных информационных объектов. Типы виртуальных информационных объектов являются типами объектов, которые не отображаются на отдельные типы объектов, но семантически представляют интерес для пользователя. Примером является тип объекта «Клиентская электронная почта», который является производным от типа объекта «Электронная почта». Этот тип объекта является «виртуальным» в том, что он не имеет конкретной схемы и, следовательно, не имеет отдельной таблицы в SMS на KIS. Вместо этого, он использует таблицу «Электронная почта» в SMS, поскольку является производным от типа объекта «Электронная почта». Хотя он не является отдельным типом объекта, пользователи будут заинтересованы в просмотре и поиске объекта «Клиентская электронная почта», как если бы он был на самом деле отдельным.

В предпочтительном варианте осуществления типы виртуальных объектов реализуются путем сохранения метаданных в соответствующей таблице в SMS (в этом случае в таблице «Электронная почта», поскольку данный тип объекта является производным от «электронной почты»). Однако разрешение запросов для такого типа объекта выполняется иным образом, отличным от регулярных запросов для отдельных типов объектов. Когда сервер SQP получает требование на семантический запрос (через XML-Web-сервис) для типа виртуального информационного объекта (такого как «Клиентская электронная почта»), он разрешает это требование путем объединения таблиц, которые совместно образуют этот тип объекта. Например, в предпочтительном варианте осуществления в случае объекта «Клиентская электронная почта» сервер будет разрешать запрос следующим SQL-подзапросом:

```
SELECT OBJECTID FROM EMAIL WHERE OBJECTID IN (SELECT OBJECTID
FROM CUSTOMERS WHERE EMAILADDRESS IN (SELECT EMAILADDRESS FROM EMAIL))
```

Этот запрос соответствует следующему: «выбрать все объекты из таблицы «Электронная почта», которые имеют значение адреса электронной почты такое, которое имеется в таблице «Клиенты». Это означает, что «клиентская электронная почта» относится к электронной почте, посланной к или от клиентов. Возможны и другие определения типа виртуального объекта, и разрешение запроса предпочтительно будет согласованным с определением. SQP предпочтительно применяет этот подзапрос ко всем запросам для «клиентской электронной почты». Этот подзапрос по существу фильтрует таблицу «Электронная почта» для выделения тех сообщений электронной почты, которые относятся к клиентам. Это возвращает желательный результат пользователю, как если бы действительно имелась таблица «Клиентская электронная почта», хотя ее реально нет.

Настоящее изобретение предусматривает разнообразие схем, связанных с каждым типом объекта. Другие схемы находятся на стадии разработки и также смогут быть использованы настоящим изобретением. Схема «Документ», например, может быть расширена полями из схемы Dublin Core: (<http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc2413.html>) и других стандартных схем. В другом примере схема "Новостная статья" может быть расширением схемы NewsML (<http://www.newsml.org>). Для примера, предпочтительная схема объекта «Пользователь», выполненная согласно изобретению, показана на фиг. 29. Все схемы предпочтительно имеют идентичный поднабор полей, подобно полям схемы «Объект». Поле MailingAddressTypeID, предпочтительно связанное со схемой объекта «Пользователь» (персона), включает элементы, показанные на фиг. 30.

Для примера схема объекта «Категория», выполненная согласно изобретению, показана на фиг. 31.

Для примера схема объекта «Документ», выполненная согласно изобретению, показана на фиг. 32.

Поле «DocumentCategory» относится к специализированной категории, которая маркирована в документе (источником данных документа), а не к семантической категории, управляемой самим KIS. Поле «DocumentFormatTypeID» относится к типу документа. Поле «PrintMediaTypeID» в предпочтительном варианте осуществления показано на фиг. 32, а поле «FormatTypeID» показано на фиг. 34.

Предпочтительный вариант схемы объекта списка сообщений электронной почты, выполненной согласно изобретению, показан на фиг. 35. Приоритеты электронной почты предпочтительно имеют значение 0, 1 и 2 соответственно низкому, среднему и высокому приоритету. Поле EmailTypeID предпочтительно включает EMAILTYPEID_EMAIL, EMAILTYPEID_NEWSPosting and EMAILTYPEID_EMAILANNOTATION (значения 1, 2 и 3). Примеры таблиц, показывающих схемы списка распространения электронной почты и объекта общедоступного каталога электронной почты в предпочтительном варианте осуществления изобретения, показаны на фиг. 36 и 37. Поле PublicFolderTypeID включает в себя элементы, показанные на фиг. 38.

Предпочтительный вариант схемы объекта события, выполненной согласно изобретению, показан на фиг. 39. Фиг. 40 показывает типы событий согласно предпочтительному варианту осуществления изобретения.

Предпочтительный вариант схемы объекта средства печати, выполненной согласно изобретению, показан на фиг. 41. Фиг. 42 показывает типы средств печати согласно предпочтительному варианту осуществления изобретения.

Фиг. 43-45 иллюстрируют дополнительные примеры того, как объекты разделяются на категории и используются согласно предпочтительному варианту осуществления изобретения. Фиг. 43 показывает типы контейнеров объектов корневого каталога. Фиг. 44 иллюстрирует иерархическую схему для определенных типов объектов. Фиг. 45 показывает собственные предикаты типов объекта «контейнер». Все типы, кроме «Персона» и «Клиент», предпочтительно наследуют все предикаты из корневого типа «Вся информация». Настоящее изобретение предусматривает для собственных предикатов типов объекта «контейнер» шаблоны, включающие, например, следующее:

Все

Новости, приводящие к прерыванию

Разделение на категории

Автор

Аннотации

Определенные связи

Вероятностные связи; популярные.

b. Компонент сбора семантических данных.

В предпочтительном варианте компонент сбора семантических данных (SDG) обеспечивает добавление, удаление, обновление записей в семантической сети через SMS. SDG содержит список ссылок на XML-Web-сервисы. Они образуют уровень абстракций информационных источников (ISAL). Каждая из этих ссылок инициализируется для сбора данных посредством адаптера источников данных (DSA). Адаптером источника данных является XML-Web-сервис, который собирает информацию от локального или удаленного источника семантических данных для заданного типа источника. Затем он возвращает XML, соответствующий записям объектов в источнике данных. Все DSA предпочтительно поддерживают один и тот же интерфейс, посредством которого SDG собирает XML-данные. Этот интерфейс использует методы для

извлечения XML-метаданных для объектов для заданного начального и конечного индекса (например, объекты 0-49);

проверки того, были ли какие-либо объекты добавлены или удалены с конкретной даты/числа (по времени DSA);

выборки XML-метаданных для объектов, которые были добавлены или удалены с конкретной даты/числа (по времени DSA);

проверки того, существует ли еще объект в источнике семантических данных - путем исследования XML-метаданных на наличие этого объекта (прошедшего как аргумент).

Если каждое обращение к XML-Web-сервису DSA не будет сопровождаться изменением состояния, то API будет включать информацию, предпочтительно в строке с командными параметрами, которая квалифицирует запрос. Например, DSA для входящей электронной почты включает такие параметры, как имя пользователя, данные входящей электронной почты которого собираются. DSA для Web-сайта или хранилища документов должен включать информацию об URL или соответствующем пути каталога.

Каждый DSA должен извлекать информацию в схеме для соответствующего типа объекта. Поскольку DSA должен быть реализован для конкретного типа объекта, SDG будет ожидать XML для схемы этого типа объекта, когда он активизирует обращение для сбора данных к DSA.

SDG обеспечивает поддержание целостности и согласованности всех таблиц баз данных в SMS (семантической сети). В этом варианте осуществления на SDG также ссылаются как на администратора семантической сети (SNM). Таблицы базы данных предпочтительно не содержат избыточных или устаревших записей. Поскольку SDG извлекает объекты с хорошо известными схемами, семантика каждого из типов объектов понятна, и SDG поддерживает согласованность таблиц соответствующим образом. Например, SDG предпочтительно не добавляет избыточных метаданных документа XML в таблицу ДОКУМЕНТЫ. SDG использует семантику документов для проверки на избыточность. В предпочтительном варианте это выполняется путем сравнения имени автора, даты/времени создания, пути к файлу и т.д. SDG также выполняет эту проверку для других таблиц (например, СОБЫТИЯ, КЛИЕНТЫ, НОВОСТИ и т.д.). Например, SDG будет выполнять проверку избыточности для событий путем исследования названия, местоположения и даты/времени. Другие таблицы поддерживаются соответствующим образом. SDG также будет обновлять объекты в таблицах базы данных, которые изменились.

SDG также реагирует на очистку таблиц базы данных. SDG предпочтительно запрашивает DSA для определения того, все ли объекты в каждой таблице, управляемой посредством DSA, еще существуют. Например, для DSA, который извлекает документы, SDG будет пропускать XML-метаданные в Web-сервис DSA и запрашивать, существует ли еще объект. DSA пытается открыть URL для данного документа. Если документ больше не существует, DSA покажет это для SDG. Отдельные DSA, но не SDG, ответственны за проверку подлинности объекта, чтобы избежать ограничений защищенности, специфических для источников данных. Например, могут иметь место ограничения защищенности, которые запрещают удаленный доступ к локальным ресурсам. В этом случае только XML-Web-сервис в DSA (который предпочтительно выполняется локально, относительно источника данных) будет иметь доступ к источнику данных. Альтернативно, некоторые DSA могут выполняться на сервере агентства, совместно с SDG и другими серверными компонентами, и дистанционно извлекать их данные.

То, что DSA осуществляют обработку подтверждения подлинности, обеспечивает дополнительную эффективность и защищенность вследствие того, что DSA исключает необходимость для SDG в знании деталей того, как открыть каждый источник данных для проверки того, существует ли еще объект. Поскольку для DSA требуется знать это (так как он извлекает XML-данные из источника данных и поэтому имеет код, специфический для этого источника данных), то обработка этой задачи является более уместной для DSA.

SDG предпочтительно поддерживает список сбора данных, который будет указывать URL для XML-Web-сервиса DSA. Администратор KIS может добавлять, удалять и обновлять записи DSA из списка сбора данных SDG. Каждый список сбора данных предпочтительно конфигурируется с учетом следующего:

1. Имя и ссылка на XML-Web-сервис DSA. Это, по существу, будет ссылаться на комбинацию из источника данных и типа объекта и на ссылку на XML-Web-сервис, который реализует DSA (например, через URL WSDL-Web-сервиса). Примеры включают следующее:

a. Microsoft Exchange 2000 Email DSA. Этот DSA будет собирать XML-метаданные электронной почты из входного почтового ящика или общедоступного каталога Microsoft Exchange 2000.

b. Microsoft Exchange 2000 Calendar DSA. Этот DSA будет собирать XML-метаданные событий из календаря Microsoft Exchange 2000.

c. Microsoft Exchange 2000 Users DSA. Этот DSA будет собирать XML-metadata для пользователей/людей из каталога Microsoft Exchange 2000.

d. Microsoft Exchange 2000 Email Distribution List DSA.

Этот SDA будет собирать метаданные списка распределения электронной почты из каталога Microsoft Exchange 2000.

e. Lotus Notes Inbox. Этот DSA будет собирать XML-метаданные из входного почтового ящика или

общедоступного каталога Lotus Notes Inbox.

f. Siebel CRM Database. Этот DSA будет собирать XML-метаданные клиентов из системы Siebel CRM.

g. Web site. Этот DSA будет собирать XML-метаданные документов с Web-сайта.

h. File Directory or Share. Этот DSA будет собирать XML-метаданные из каталога файлов или совместно используемого ресурса.

i. Saba E-Learning LMS Repository. Этот DSA будет собирать XML-метаданные электронного обучения из хранилища Системы Управления Обучением Saba (LMS).

j. Microsoft Sharepoint Document DSA. Этот DSA будет собирать XML-метаданные из серверного рабочего пространства Microsoft Sharepoint.

k. Reuters News Repository. Этот DSA будет собирать XML-метаданные новостных статей из хранилища новостных статей Reuters.

2. Описание записи собранных данных DSA.

3. Строка, указывающая информацию инициализации для DSA.

4. График сбора данных - показывает, как часто SDG должен «ползать» по DSA, чтобы собрать XML-метаданные.

В предпочтительном варианте агентство инициализируется с доменом пользовательского каталога и именем группы. В этом случае SDG предпочтительно автоматически вводит запись списка сбора данных для каталога пользователей DSA. Например, если агентство конфигурировано с пользовательским каталогом Exchange 2000 с доменным именем "Foo" и адресной книгой или групповым именем "everyone", то SDG создает запись в списке сбора данных с DSA пользователей в Exchange 2000 (инициализируется с этими параметрами). Альтернативно, агентство может быть конфигурировано для получения своего пользовательского каталога от любого сервера приложений электронной почты (Microsoft Exchange или Lotus Notes). SDG инициализирует записи списка сбора данных с DSA входящей почты и календаря для системных пользователей (и агентом знаний электронной почты, описано ниже). Эти три DSA записей в списке сбора данных (пользователи, входящая почта и календарь) инициализируются по умолчанию. Входящая почта предпочтительно используется для сохранения электронных почтовых отправок и аннотаций агентства, а DSA календаря используется для сохранения событий, посланных в агентство пользователями. Другие заказные агенты (DSA) могут быть добавлены администратором агентства.

SDG также отслеживает последнее время, когда SDA сообщил ему о добавлении или удалении объектов к/из источника данных. Эта информация даты/времени предпочтительно базируется на часах. Каждый раз, когда SDA сообщает о наличии новых данных или об удалении данных, SDG будет обновлять информацию даты/времени в своей записи для SDA и собирать всю новую или удаленную информацию в SDA. SDG затем будет обновлять таблицы базы данных.

SDG предпочтительно отображает XML-информацию, которую он получает от SDA, в семантической сети настоящего изобретения. SDG сохраняет все XML-метаданные в таблицах базы данных в SMS. Кроме того, SDG осуществляет синтаксический анализ XML, принятых от SDA, и, если необходимо, отображает семантические связи к конкретным XML-полям. SDG добавляет или обновляет семантические связи в случаях, когда XML включает информацию, которая «связывает» объекты вместе. Например, схема для объекта электронной почты включает поля, в том числе "From" («от»), "To" («к»), "Cc" («точная копия»), "Bcc" («скрытая копия») и "Attachments" («приложения»). В случае столбцов "From", "To", "Cc", "Bcc" поля в XML ссылаются на адрес электронной почты (отделяются разделителями, такими как «;», «или» или пробелом). В случае столбца "Attachments" это поле будет ссылаться на пути к файлу или поля, которые присоединяются к сообщению электронной почты (отделяются разделителями, такими как «,»). Этот исходный XML сохраняется в таблице EMAIL (ЭЛЕКТРОННАЯ ПОЧТА) базы данных, вместе с другими столбцами. Кроме того, SDG осуществляет синтаксический анализ полей объекта электронной почты и добавляет семантические связи к другим объектам, которые идентифицированы содержимым этих полей. Например, если поле "to" содержит john@foo.com, а поле "attachments" содержит строку "c:\foo.doc, c:\bar.doc", то SDG будет обрабатывать электронную почту следующим образом:

1. Найти любой объект в таблице USERS (ПОЛЬЗОВАТЕЛИ) с адресом электронной почты john@foo.com. Также осуществить поиск других объектов пользователь с адресами электронной почты в полях FROM, TO, CC, BCC.

2. Если какие-либо объекты найдены, добавить запись семантической связи в таблицу SEMANTICLINKS (СЕМАНТИЧЕСКИЕ СВЯЗИ) с ID объекта электронной почты в качестве субъекта и с соответствующим ID типа предиката. В этом случае предикат PREDICATETYPEID_CREATOR относится к отправителю сообщения электронной почты. Предикат PREDICATETYPEID_SENTTO используется для связи объекта электронной почты и объектов ПОЛЬЗОВАТЕЛЬ, на которые ссылаются посредством содержимого поля "to" в XML-метаданных электронной почты. Предикат PREDICATETYPEID_COPIEDTO и PREDICATETYPEID_BLINDCOPIEDTO используются для связи объектов в полях "cc" "bcc".

В случае приложений SDG выделяет XML-метаданные из присоединенных документов. Если XML-объект с путем к файлу уже существует в SMS (или, иными словами, в семантической сети), то SDG будет обновлять метаданные. Если XML-объект не существует, то SDG создает новый объект документа с XML-метаданными. SDG будет добавлять запись в таблицу SEMANTICLINKS с ИД объекта электронной почты в качестве субъекта, ИД объекта нового документа в качестве субъекта и предикатом PREDICATETYPEID_ATTACHEDTO. Это позволяет пользователю иметь возможность перемещаться от сообщения электронной почты к его приложениям и затем использовать приложения в качестве опорного пункта для продолжения просмотра семантической сети, например, с использованием семантических инструментальных средств, подобных интеллектуальной лупе (см. ниже).

SDG не создает никаких объектов в событии, для которого он не находит объектов пользователя, которые согласованы с записями в XML-полях. Предпочтительно SDG собирает информацию из SDA каталога, когда пользователь вручную добавлен к агентству. Администратор агентства предпочтительно добавляет пользователей к агентству через группу пользователей по свойствам агентства.

Следующее иллюстрирует пример отображения исходных XML-метаданных электронной почты на семантическую сеть.

```
<emailfrom=john@foo.com to="nosa@nervana.net"
cc="steve@nervana.net" bcc="patrick@nervana.net"
subject="Meeting this Friday" body="Let us meet on Friday at 2pm" attachments="c:\foo.doc;
c:\bar.htm">
</email>
```

На фиг. 46 представлено преобразование этого объекта электронной почты в графы объекта.

c. Компонент контроля согласованности семантической сети.

Компонент контроля согласованности семантической сети (СС) дополняет проверку согласованности, которая выполняется посредством SDG. Как описано выше, SDG поддерживает целостность таблиц базы данных путем запрета добавления избыточных записей в семантическую сеть (из разных источников данных). СС также гарантирует согласованность таблиц ОБЪЕКТЫ и СЕМАНТИЧЕСКИЕ СВЯЗИ. СС периодически проверяет таблицу ОБЪЕКТЫ, чтобы убедиться, что каждый объект существует в собственной таблице (предпочтительно путем проверки значения поля OBJECTID («идентификатор объекта»). Например, запись объекта документа в таблице ОБЪЕКТЫ предпочтительно также существует в таблице ДОКУМЕНТЫ (с тем же самым ИД объекта). СС удаляет любой объект в таблице ОБЪЕКТЫ без соответствующего объекта в собственной таблице (ДОКУМЕНТЫ, СОБЫТИЯ, ЭЛЕКТРОННАЯ ПОЧТА и т.д.) и наоборот.

СС также несет ответственность за поддержание согласованности таблицы СЕМАНТИЧЕСКИЕ СВЯЗИ. Семантикам этой таблицы предпочтительно следующая: семантическая связь не может существовать, если ее субъект («связь от») или ее объект («связь к») не существует. Например, если объект А связан с объектом В с предикатом Р, и если либо А, либо В удален, то связь должна быть удалена. СС периодически проверяет таблицу СЕМАНТИЧЕСКИЕ СВЯЗИ. Если какой-либо из субъектов или объектов удален, то СС удаляет запись семантической связи.

Проверки согласованности могут быть реализованы в коде в самом KIS или как сохраненные процедуры или ограничения на уровне базы данных.

d. Машина логического вывода.

Машина (механизм) логического вывода обеспечивает добавление семантических связей к семантической сети. Машина логического вывода использует правила логического вывода, которые состоят из набора эвристических правил, для добавления семантических связей на основе действующей семантической активности. Машина логического вывода предпочтительно может удалять семантические связи. Агенты решения (описаны ниже) используют машину логического вывода для содействия специалистам в области информационных технологий в принятии решений.

Машина логического вывода работает путем разработки семантической сети и добавления новых семантических связей, которые основаны на вероятностных логических выводах. Например, машина логического вывода предпочтительно контролирует семантическую сеть и наблюдает шаблоны (образцы, конфигурации), в которых посылается электронная почта, тип посланной электронной почты и кем послана. Машина логического вывода логически выводит из этой информации «фоновую» (вводную) информацию, например, экспертный анализ пользователя, связанный с различными категориями сущности в сфере действия контроля машины логического вывода. Например, машина логического вывода добавляет семантические связи с предикатом PREDICATETYPEID_EXPERTON, чтобы показать, что пользователь является экспертом в конкретной категории. Субъект в этом случае будет объектом пользователя, а объект будет объектом категории. Для логического вывода на этой основе машина логического вывода предпочтительно конфигурируется для наблюдения семантической активности, по меньшей мере, для определенного периода времени (например, две недели) или только для логического вывода связей после того, как пользователь послал, по меньшей мере, некоторое предварительно определенное количество сообщений или создал некоторое количество документов. Машина логического вывода логически выводит новую связь путем отслеживания статистики на связях PREDICATETYPEID_CREATOR (ИД типа

предиката - создатель) и PREDICATETYPEID_CONTRIBUTOR (ИД типа предиката - участник).

Например, машина логического вывода может логически вывести, что пользователи являются экспертом по некоторой категории, если

из всех категорий сообщений электронной почты, написанных ими, эта категория является одной из N верхних (конфигурируемо);

они писали сообщения электронной почты по той же самой категории в среднем M раз или более за неделю (конфигурируемо);

они писали по меньшей мере O сообщений электронной почты (конфигурируемо) в последние 3 месяца.

Более сложные модели логических выводов, позволяющие точно логически вывести эти данные, также могут быть использованы. Например, могут использоваться распределения вероятности, а также модели статистической корреляции. Предпочтительно эти модели будут разрабатываться на базе отдельных сценариев с течением времени.

Машина логического вывода также несет ответственность за удаление связей, которые она могла добавить. Например, если служащий сменил работу, то он может перестать быть экспертом по конкретной категории (относительно других служащих). Как только машина логического вывода обнаруживает это (например, путем просмотра шаблонов электронной почты), она удаляет семантические связи, которые указывают, что данное лицо является экспертом по некоторой категории.

Логически выведенные семантические связи являются важными для сценариев, которые связаны с вероятностными семантическими запросами. Например, в варианте осуществления изобретения, использующем информационного агента, пользователи могут перетащить и оставить документ из их файловой системы в агента (например, People.Research.All - Люди.Исследование.Все). В этом случае пользователям будет желательно знать специалистов в исследовательском отделе, которые являются экспертами по документу. Браузер затем вызовет SQML-запрос с агентом в качестве ресурса (или субъекта), с предикатом `person:experton`, и путем документа в качестве объекта. Презентатор затем извлечет XML-метаданные для документа и вызовет XML-Web-сервис, имеющийся в агентстве, который содержит данного агента, с предикатом ИД и XML-метаданными в качестве аргумента. Серверный процессор семантических запросов в агентстве обрабатывает этот вызов XML-Web-сервиса и преобразует вызов в SQL-запрос, согласованный с моделью данных семантической сети. В этом примере вызов предпочтительно разрешается следующим образом.

1. Для всех записей семантической предметной области в KDM вызвать соответствующий KBS для категоризации документа.

2. Отобразить возвращенные категории на объекты категории в семантической сети (путем сопоставления URL).

3. Активизировать запрос с использованием запроса агента People.Research.All в качестве подзапроса.

В этом примере окончательный запрос будет иметь вид: `SELECT*FROM USERS WHERE DEPARTMENT LIKE "RESEARCH" AND OBJECTID`

`IN (SELECT OBJECTID FROM SEMANTICLINKS WHERE OBJECTTYPEID = 32 AND PREDICATETYPEID = 98 AND SUBJECTID IN (SELECT OBJECTID AS SUBJECTID FROM CATEGORIES WHERE OBJECTID IN (34, 56, 78)) AND LINKSCORE > 90)`

Этот запрос предполагает, что ИД типа объекта для типа объекта пользователя есть 32, значение ИД типа предиката для PREDICATETYPEID_EXPERTON есть 98, документ принадлежал к категориям с ИД объекта 34, 56 и 78 и что порог оценки семантической связи есть 90.

е. Серверный процессор семантических запросов.

Серверный процессор семантических запросов (SQP) реагирует на семантические запросы от клиентов KIS. SQP является предпочтительно главным пунктом входа в семантическую сеть на KIS (или Агентстве). SQP подвергается воздействию через XML-Web-сервис агентства. Процессы SQP управляют семантическими запросами агента и обобщенными (клиентскими) семантическими запросами с фильтрами семантических связей. Для запросов с фильтрами серверного агента информационный агент пропускает имя агента аргументы индекса объекта на активизируемый SQP. Например, браузер может запросить объекты 0-24 на агенте Documents.Technology.Wireless.All. В этом примере SQP отыскивает запрос агента в таблице «Агенты» и запускает запрос в базу данных, которая содержит хранилище семантических метаданных (SMS). Запрос агента предпочтительно сохраняется как SQL или в другом хорошо известном формате, таком как XQuery или XQL. SQP может преобразовывать формат запроса в формат, который понятен базе данных (которая содержит все таблицы). Ввиду того, что большинство коммерческих баз данных понимают формат SQL, он предпочтительно будет действовать как установленный по умолчанию формат запроса агента.

Запрос агента предпочтительно следует правилам запроса, изложенным выше. Поэтому запрос возвращает ИД объекта, а не поля схемы для типа объекта агента. В вышеописанном примере, Documents.Technology.Wireless.All активизирует запрос агента "SELECT OBJECTID FROM DOCUMENTS WHERE...". SQP обеспечивает выдачу запроса, который фильтруется запросом агента, но который воз-

вращает действительные метаданные для типа объекта (в этом случае тип объекта «документ»). В этом случае запрос будет иметь следующий вид:

```
SELECT * FROM DOCUMENTS WHERE OBJECTID IN (SELECT OBJECTID FROM DOCUMENTS WHERE ...)
```

Этот запрос возвращает столбцы данных для схемы «документ» для всех объектов с ИД объекта, который согласуется с первоначальным запросом агента. SQP просматривает результаты метаданных запроса базы данных и преобразует их в правильно сформированный XML с использованием соответствующей схемы для типа объекта агента (в данном случае «документ»). В случае, когда база данных поддерживает извлечение исходного XML, SQP оптимизирует запрос путем запроса в базу данных о выдаче результатов в формате XML. Это обеспечивает лучшие характеристики, поскольку SQP не должен выполнять дополнительный этап преобразования. SQP пропускает XML назад к запрашивающему пользователю через XML-Web-сервис агентства.

SQP предпочтительно обрабатывает более сложные запросы, которые пропускаются семантическим браузером (или иным клиентом XML-Web-сервиса). Например, такие запросы могут принимать форму следующих интерфейсов программирования приложений (API) XML-Web-сервиса:

```
String
InvokeSemanticQuery(
Integer BeginIndex,
Integer EndIndex,
String AgentName,
Integer NumberOfLinks,
String OperatorNames[],
String LinkPredicateNames[],
String LinkTypeNames[],
String LinkObjects[]).
```

В этом примере символ "[]" относится к массивам. API принимает индекс начала с отсчетом от нуля, индекс конца с отсчетом от нуля, факультативное имя агента, целое число, указывающее число семантических связей, массив имен операторов, массив имен предиката связи, массив имен типа связи и массив строк, которые относятся к объектам связи. Если имя агента NULL (" "), то SQP обрабатывает запрос "как он есть", без какого-либо предварительного фильтра агента. Это будет иметь место для запросов, которые полностью генерированы клиентом. Массивы имеют переменные размеры, так как параметр "NumberOfLinks" (количество связей) указывает размер каждого массива. Имя оператора включает действительные предварительно определенные операторы, включая логические операторы, которые могут быть использованы для формулирования запросов в SQL или других форматах запроса. Примеры включают term:or (оператор «или») and term:and (оператор «и»). Имена предиката связи могут включать один или более предварительно определенных предикатов (например, term:relevantto («релевантен»), term:reportsto («сообщает»), term:isentto («послан»), term:annotates («аннотирует»), term:annotatedby («аннотирован посредством»), term:withcontext («с контекстом») и т.д.). Имена типа связи указывают на тип связи объектов. Обычные примеры включают: term:url и term:object. В случае term:url строка объекта связи относится к правильно сформированному URL (универсальному указателю ресурса), содержащему объекты://... или агента://.... В случае term:object, аргумент будет представлять собой правильно сформированную команду XML-метаданных, относящуюся к объекту, определенному в настоящем изобретении. Этот объект предпочтительно разрешается (отображается) клиентом или другим агентством. API возвращает строку, которая содержит XML-результаты (помимо возвращенного значения для самого вызова метода XML-Web-сервиса). Например, запрос в SQL-формате со следующими данными:

```
<resource type="term:url"
Agent://all.criticalpriority.all@abc.com/Agency.asp> <link predicate="term:relevantto"
type="term:,object" object://4576>
<link operator="or"
predicate="term:intersects"
type="term:url"
Agent://email.wireless.all@abc.com/Agency.asp>
</resource>
```

разрешается на агентстве, находящемся на Web-сервисе на abc.com/Agency.asp, в следующий вид:

```
InvokeSemanticQuery(
0,
24,
"all.criticalpriority.all",
2,
{"term:and", "term:or"},
{"term:relevantto", "term:intersects"},
{"term:object", "term:url"},
```

```
{"object://4576","Agent://email.wireless.all@abc.com/Agency.asp"}).
```

Это предпочтительно разрешается в SQL-запрос следующего вида:

```
SELECT TOP 25 * OBJECTS WHERE OBJECTID IN (SELECT OBJECTID FROM OBJECTS WHERE CREATIONDATETIME='02/26/02' AND (OBJECTID [RELATEDTO] [OBJECT WITH ID 4576]) AND OBJECTID IN (SELECT OBJECTS FROM EMAIL WHERE CATEGORY [IS] 'WIRELESS'))
```

Пример SQL-запроса использует стенографию для иллюстрации типа запроса, который будет генерироваться посредством SQP. SQP извлекает XML и возвращает его запрашивающему пользователю. Этот XML представлен в форме SRML (или язык разметки семантических результатов), что является определением XML-метасхемы для результатов семантического запроса в предпочтительном варианте осуществления изобретения.

Пример А, показанный в приложении, является образцом буфера или документа семантических результатов на языке SRML. Это образец XML, который агентство возвращает в ответ на семантический запрос. Клиентская поверхность (Skin) принимает эти результаты и генерирует форму их представления (используя XSLT и/или сценарий), на основе свойств поверхности и агента (поверхность/контекст поверхности/элемент сопряжения поверхности объекта), размера располагаемой области дисплея, соображений невозможности реализации и других признаков поверхности.

f. Синтаксический анализатор естественного языка.

Синтаксический анализатор естественного языка (NLR) предпочтительно преобразует текст на естественном языке либо в вызов API, который понимает процессор SQP, либо в исходный SQL (или подобный формат запроса), который может обрабатываться базой данных. Синтаксический анализатор естественного языка пропускает текст непосредственно с семантического браузера или по электронной почте посредством агента знаний электронной почты (см. ниже).

g. Агент знаний электронной почты.

Сервер интеграции знаний (СИЗ, KIS) предпочтительно включает в себя один главный компонент публикации, называемый агентом знаний электронной почты (или информационным агентом предприятия (EIA)). Этот агент функционирует, по существу, как цифровой служащий (сотрудник) и предпочтительно включает в себя уникальный адрес электронной почты (например, специализированное имя, выбранное администратором агентства). Агент знаний электронной почты образует дополнение к существующим издательским инструментальным средствам, таким как Microsoft Office, SharePoint и т.д., путем добавления метода "Fire and Forget" («запустить и забыть») публикации информации и совместного использования знаний. Это особенно полезно в случаях, когда лицо, публикующее информацию, не знает, кому она может оказаться интересной.

В предпочтительном варианте осуществления изобретения пользователи посылают электронную почту агенту знаний электронной почты для публикации комментариев, аннотаций, документов, приложений и т.д. Агент знаний электронной почты извлекает значение (смысл) из документов электронной почты и надлежащим образом добавляет их в семантическую сеть. Другие пользователи могут получить доступ к публикуемой информации через агентов или другие инструментальные средства платформы, такие как "drag and drop" (операция графического интерфейса «перетащить и оставить»), интеллектуальная лупа и т.д. (описано ниже).

Агент знаний электронной почты является системным компонентом, который создан администратором агентства. Имя системного пользователя указывается, когда сервер впервые инсталлирован. Системный пользователь предпочтительно соответствует пользователю электронной почты в системе электронной почты предприятия (например, Microsoft Exchange, Lotus Notes и т.д.). В этом варианте осуществления агент электронной почты имеет свой собственный почтовый ящик, календарь, адресную книгу и т.д. Эти элементы в свою очередь соответствуют объектам на сервере электронной почты для системного пользователя. При инсталляции сервера KIS инсталлирует соответствующий DSA (агент системного каталога) для входящей почты системы (в зависимости от приложения электронной почты). KIS предпочтительно автоматически добавляет запись в списке компонента сбора данных в SDG, указывая, что входящая почта система должна периодически просматриваться для сбора данных электронной почты.

Поскольку агент знаний электронной почты является первоклассным адресом электронной почты, он также служит источником уведомлений и источником запросов (для непосредственной передачи сообщений на естественном языке). Уведомления от агентства предпочтительно посылаются агентом знаний электронной почты (указывая, что имеется новая релевантная информация, в которой пользователь может быть заинтересован, и т.д.). Агент знаний электронной почты также может принимать электронную почту от пользователей в виде запросов на естественном языке. Эти сообщения синтаксически анализируются посредством SQP и обрабатываются. XML-результаты предпочтительно посылаются пользователю в виде XML-файла (с соответствующей установленной по умолчанию поверхностью), генерируемого с XSLT, обработанным по XML-результатам запроса на естественном языке.

Поскольку агент знаний электронной почты является хорошо известным компонентом или «сотрудником», администратор агентства предпочтительно добавляет адрес в списки распространения. Этот этап позволяет компоненту сбора информации SDG семантически индексировать все сообщения элек-

тронной почты в этих списках распространения, тем самым наполняя семантическую сеть путем непрерывной интеграции агента знаний электронной почты в списки распространения, полезные для пользователей. Это является в значительной степени непрерывным («бесшовным») методом интеграции информационной нервной системы настоящего изобретения с обычным методом работы людей в организации.

Аннотации. Агент знаний электронной почты предпочтительно используется для публикации аннотаций. В настоящем изобретении аннотации предпочтительно являются сообщениями электронной почты. В предпочтительном варианте тип объекта аннотации является подклассом типа объекта электронной почты. Это позволяет пользователям использовать электронную почту, в типовом случае самое обычное инструментальное издательское средство, для аннотирования объектов в семантическом браузере. Пользователи могут аннотировать объекты и добавлять приложения к аннотациям. Эти приложения семантически индексируются посредством SDG на KIS. Это делает возможным сценарии, где пользователь может перемещаться от документа, например, к аннотации, к приложению к этому документу, к статье Reuters, к событию в промышленности, которое состоится на следующей неделе.

Процесс, описанный для семантической индексации электронной почты (путем отображения XML-схемы электронной почты на семантическую сеть), также применим и для аннотаций. Однако в случае аннотаций в предпочтительном варианте осуществления изобретения желательная дополнительная обработка. Более конкретно, когда пользователь выполняет щелчок на опции «Аннотировать» на объекте в окне презентатора в семантическом браузере (описано ниже), браузер загружает зарегистрированного клиента электронной почты на локальную машину (Microsoft Outlook, Microsoft Outlook Express и т.д.). Поле "to" («кому») заполнено адресом системного пользователя для агентства, где находится объект. Поле субъекта заполнено специальной строкой, например, "annotation: object=[objectid]" («аннотация: объект=[ИД объекта]). SDG извлекает новые XML-метаданные электронной почты из DSA путем получения события, или в следующий раз он запрашивает от SDA дополнительные данные. В предпочтительном варианте осуществления этот процесс опроса происходит часто. DSA возвращает XML-метаданные объекта электронной почты, не обращая внимания на то, что объект электронной почты ссылается на тип объекта электронной почты или на тип объекта аннотации. SDG обрабатывает XML-метаданные электронной почты и анализирует поле "subject". Если SDG «видит» префикс "annotation:", то он знает, что электронная почта в действительности является аннотацией, и переходит к извлечению аргумента ИД объекта из текста субъекта. SDG обновляет семантическую сеть остальными сообщениями электронной почты (добавляя каждое сообщение в таблицы ОБЪЕКТЫ и ЭЛЕКТРОННАЯ ПОЧТА, добавляя семантические связи для полей "from," "to," "cc," "bcc" и "attachments", где необходимо, и т.д.). В предпочтительном варианте осуществления SDG выполняет дополнительный этап. В частности, он добавляет запись семантической связи, которая связывает объект электронной почты с объектом, указанным в аргументе ИД объекта в тексте субъекта (с предикатом PREDICATETYPEID_ANNOTATES).

В настоящем изобретении аннотация обрабатывается как другая семантическая связь со специальным предикатом. В результате все семантические признаки применяются к аннотациям, такие как семантическая навигация по семантическим связям, семантические запросы и т.д. Например, пользователь может запросить все аннотации, написанные кем-либо из его подразделений за последние шесть месяцев. Это может быть выполнено в семантическом браузере путем «перетаскивания», например, опции Annotations.All агента поверх опции People.MyTeam.All и затем сортировки результатов или путем создания интеллектуального агента, который, в свою очередь, активизирует мастера «Создать интеллектуального агента» для создания запроса.

h. Администратор предметной области знаний (KDM).

Администратор предметной области знаний (KDM) является компонентом на сервере KIS, ответственным за дополнение и поддержание относящейся к предметной области информации в семантической сети. KDM, по существу, «аннотирует» семантическую сеть информацией предметной области. Инициализируется с URL, связанным с одним или несколькими экземплярами сервера базы знаний (KBS), который, в свою очередь эффективно хранит «знания» для одной или более семантических областей. KBS имеет онтологию и категории соответственно таксономии для каждой семантической области, которую он поддерживает. Кроме того, агент в семантической области (соединенный с KBS) реагирует на семантические запросы. Если агент не принадлежит к семантической области, он не может соответствовать семантическим запросам (которые требуют онтологии или таксономии). Вместо этого он только отвечает на запросы, основанные на ключевых словах (хотя он еще будет обеспечивать услуги контекстно- и времязависимого поиска, однако, доступные контексты будут ограничены).

Каждая запись в KDM является записью семантической области. Запись семантической области имеет URL в KBS и имя семантической области. Имя семантической области отображается на конкретную топологию на KBS. В предпочтительном варианте настоящего изобретения имена семантической области соблюдают следующее соглашение:

<Top Level Domain Name>\<Secondary Level Domain Name> (Имя области верхнего уровня/имя области вторичного уровня)

Примерами имен семантических областей могут служить следующие:

Industries (промышленность)

Industries\Pharmaceuticals\LifeSciences (промышленность/фармацевтика/науки о жизни)
 Industries\InformationTechnology (промышленность/информационные технологии)
 General\Sports.Basketball\NBA (Общее/Спорт.Баскетбол/NBA)
 General\Sports.Basketball\CBA (Общее/Спорт.Баскетбол/CBA)

Альтернативно имена семантических областей могут определяться как «доменные пути», если они полностью определены. Полное определение реализуется путем добавления префикса доменного имени Интернет к началу пути. Это указывает на «владельца» или «источник» семантической области. Например, "Nervana.NET\Industries\Pharmaceuticals" ссылается на семантическую область "Industries\Pharmaceuticals" соответственно интернетовскому доменному имени "NERVANA.NET". В другом примере "Reuters.com\Sports\Basketball" ссылается на "Sports\Basketball" в "Reuters.com." Используя этот метод, можно обеспечивать глобальную поддержку уникальных доменных имен и путей.

Администратор предметной области знаний (KDM) периодически запрашивает каждый KBS в своем списке записей предметных областей категории в области знаний. KDM предпочтительно реализован как XML-Web-сервис на сервере интеграции знаний KIS. KDM включает в себя опции конфигураций для каждой записи семантической области. Одна из этих опций может включать в себя график, в соответствии с которым KDM будет обновлять семантическую сеть информацией конкретной предметной области в соответствии с записью семантической области.

Например, администратор агентства может конфигурировать KDM (через KIS) для просмотра семантической области на KBS каждый день в 1 ч дня. График обновления должен быть согласованным с тем, как часто, по мнению администратора, происходят изменения в онтологии или таксономии на KBS.

KIS предпочтительно активизирует KDM периодически и запрашивает его об обновлении таблиц КАТЕГОРИИ. В предпочтительном варианте KDM вызывает KBS (через вызов API XML-Web-сервиса) для получения обновленных категорий для имен семантической области в записи семантической области, что соответствует конкретной таксономии. Пример вызова API может быть следующим: GetCategoriesForSemanticDomain (String SemanticDomainName) (получить категории для семантической области (строка имени семантической области)). KBS возвращает список на основе XML всех категорий в семантической области, на которые ссылается имя семантической области. Этот XML-список согласуется со схемой КАТЕГОРИИ, показанной выше (URL категории, имя, описание, URL KBS и имя семантической области).

KDM обновляет таблицу КАТЕГОРИИ этой информацией. Для записей категорий, которые уже имеются в таблице, KDM обновляет имя и описание. Для новых записей KDM запрашивает новый ИД объекта от администратора объекта и присваивает его записи категории. Поскольку в предпочтительном варианте осуществления категория является «объектом», она унаследует от типа объекта и поэтому имеет ИД объекта.

KDM синхронизирует таблицу КАТЕГОРИИ со списком КАТЕГОРИИ на KBS (для конкретной семантической области) путем удаления записей в таблице КАТЕГОРИИ, отсутствующих в новом списке, после анализа URL записей категорий и получения релевантного URL для KBS и имени семантической области. Если запись семантической области удалена из KIS, то KDM удаляет все записи категорий с соответствующим именем семантической области и URL для KBS. По существу, это будет подобно просеиванию (сортировке) агентства на имеющиеся знания.

KDM осуществляет периодическую категоризацию всех «объектов знания» в семантической сети на основе своих записей семантических областей. Когда новые объекты добавляются к семантической сети посредством SDG, SDG запрашивает, чтобы KDM осуществил категоризацию объектов. KDM перечисляет все экземпляры KBS в своих записях семантических областей и активизирует вызовы XML-Web-сервиса с XML объекта в качестве аргумента. В предпочтительном варианте осуществления KBS возвращает результат в XML-буфере, подобный следующему:

```
<results>
  <result categoryurl="category://foo"
  score="91"> <result categoryurl="category://bar"
  score="93">
  <result categoryurl="category://foobar" score="100">
</results>
```

Эта информация указывает веса семантической категоризации XML-объекта для категорий в семантической области на KBS. В предпочтительном варианте осуществления настоящего изобретения запись семантической области инициализируется с порогом (0-100), указывающим минимальный вес, который KDM может запросить от KBS. KBS возвращает оценки, которые превышают предварительно определенный порог. KDM аннотирует семантическую сеть на основе этих результатов категоризации. Это предпочтительно выполняется путем добавления или обновления семантической связи с ИД типа предиката «принадлежит к категории» с ИД объекта категории в результате. KDM затем обновляет таблицу СЕМАНТИЧЕСКИЕ СВЯЗИ. Предположим, например, что категоризованный объект имеет значение 56 своего ИД объекта, тогда запрос обновления будет иметь вид:

```
UPDATE SEMANTICLINKS SET LINKSCORE=91WHERE OBJECTID=56
```

AND PREDICATETYPEID=67 AND SUBJECTID IN (SELECT OBJECTID AS SUBJECTID FROM CATEGORIES WHERE URL LIKE "CATEGORY://FOO")

KDM периодически сканирует и присваивает категории всем «объектам знаний» (документам, новостям, статьям, событиям, электронной почте, предпочтительно не включая объекты, подробные людям). Этот процесс предпочтительно возникает, даже если объект в семантической сети, ранее категоризованный как KBS, может стать «более интеллектуальным» и поэтому обеспечивает более совершенную категоризацию. В таком случае результаты могут изменяться, даже если повторен тот же самый запрос категоризации. Это будет происходить, например, если онтология на KBS обновлена. Таким образом, в предпочтительном варианте категоризация будет выполняться как в случае, когда объект добавлен к семантической сети с помощью компонента сбора семантических данных, так и периодически, чтобы гарантировать, что семантическая сеть содержит самые обновленные знания предметной области.

i. Другие компоненты.

Администратор предпочтительных агентов. В агентствах, которые поддерживают пользовательские состояния, администратор предпочтительных агентов управляет списком предпочтительных для каждого пользователя агентов. В предпочтительном варианте осуществления администратор предпочтительных агентов сохраняет отображение пользовательских имен на предпочтительных агентов в таблице предпочтительных агентов пользователя.

Администратор составного агента. Администратор составного агента управляет созданием, удалением и обновлением составных агентов. Как описано выше, составные агенты это такие агенты, которые состоят из других агентов в системе и инициализируются для возврата объединения или пересечения результатов запроса в содержащихся агентах. Администратор составного агента управляет всеми составными агентами в системе и отображает составных агентов на агентов, которые они содержат посредством таблицы отображения составных агентов.

Администратор составного агента представляет функции для создания составных агентов, исключения, переименования, добавления и удаления из них агентов и указания желателно ли объединение или пересечение. Составные агенты могут добавляться к другим составным агентам. При активизации процессор семантических запросов запрашивает у администратора составного агента о своем составном запросе. Администратор составного агента перемещается по своему графу отображения агентов и возвращает комплексный запрос из всех запросов всех агентов, которые он содержит. Если агенты удалены, составные агенты «принимают» новое состояние, когда они вызываются, игнорируя запрос агента. Иными словами, составление запросов делается только для агентов, которые продолжают существовать. Если составленный агент обнаруживает, что один из его агентов удален, то он удаляет запрос из своего отображения.

Администратор пользовательского профиля (UPM). Администратор пользовательского профиля (UPM) предпочтительно использует машину логического вывода для логического вывода пользовательского профиля на текущей основе. UPM комментирует для семантической сети на основе обратной связи от пользователей об их явно выраженных предпочтениях. В предпочтительном варианте осуществления этот процесс связан с использованием предиката PREDICATEID_ISINTERESTEDIN (ИД предиката «заинтересован в»). UPM логически выводит семантические связи и создает комментарий для семантической сети с предикатом PREDICATEID_ISLIKELYTOBEINTERESTEDIN (ИД предиката «вероятно, заинтересован в»). Все результаты запросов для пользователя будут квалифицироваться (вне полосы) с запросом в семантическую сеть для вышеуказанного предиката PREDICATEID_ISLIKELYTOBEINTERESTEDIN. Результаты запроса базируются на привычках пользователя, по мере того как машина логического вывода изучает их с течением времени.

Альтернативно UPM может быть конфигурирован с информацией пользовательского профиля, сохраненной в хранилище пользовательских состояний (USS). Эта информация вручную вводится у клиента, указывая пользовательские предпочтения. Эта информация переносится и сохраняется на сервере, с которым взаимодействует пользователь. Эти предпочтения связаны с различными схемами. Например, для документов схема может основываться на предпочтительных категориях. Для сообщений электронной почты схема может базироваться на предпочтительных категориях, авторах или приложениях. Это два из множества возможных примеров. UPS создает комментарий для семантической сети на основе информации, вручную введенной в USS.

Серверный администратор уведомлений (SNM). Серверный администратор уведомлений (SNM) обеспечивает пакетирование серверных уведомлений и направление их пользователям. В предпочтительном варианте осуществления пользователи регистрируются на получение серверных уведомлений на уровне агента. Каждый агент имеет возможность запускать уведомления о результатах своих запросов. Серверный администратор уведомлений определяет, каким образом фильтровать результаты запросов и форматировать их для доставки по электронной почте, голосовой связи, на пейджер или с помощью иного механизма уведомления, например услуги уведомления Microsoft .NET Alerts. Серверный администратор уведомлений поддерживает информацию о времени последнего считывания уведомления пользователем. SNM предпочтительно только уведомляет пользователя, когда имеется новая информация для конкретного пользователя у агента с момента последнего считывания.

Обнаружение агента. С использованием процедуры обнаружения агента на основе многоадресной передачи каждый агент посылает многоадресные оповещения, указывающие на их присутствие в локальной многоадресной сети. Администратор агентства устанавливает многоадресный параметр TTL (предписанное время жизни пересылаемого пакета). Настоящее изобретение предпочтительно использует протокол SAP (протокол оповещения о сеансе) с хорошо известным портом 9875 и TTL равным 255 или специализированный порт оповещений с настраиваемым TTL. Детальная информация о протоколе SAP содержится в следующем источнике:

<http://sunsite.cnlabswitch.ch/ftp/doc/standard/rfc/29xx/2974>, включенном в настоящее описание посредством ссылки.

Информационный агент предпочтительно включает в себя компонент приемника, который принимает оповещения протокола SAP. В предпочтительном варианте осуществления оповещения посылаются как XML и будут включать в себя следующую информацию:

ИД сервера (это уникальный идентификатор)

URL (универсальный указатель ресурса) сервера (это HTTP URL к XML-Web-сервису агентства)

Период оповещения (Т) - указывает время между каждым оповещением

Имеются ли какие-либо новые агенты в агентстве с момента последнего оповещения, и время создания последнего агента (по часам агентства).

Каждое агентство посылает XML-оповещение и использует прямое исправление ошибок (FFC) или прямое исправление пропусков для кодирования каждого пакета. Это делает систему устойчивой к пропуску пакетов. Альтернативно агентство может быть конфигурировано для отправки XML-оповещений несколько раз последовательно (за оповещение).

Многоадресный приемник информационного агента предьявляет подобную каталогу семантику администратору семантической среды. Приемник агрегирует все XML-оповещения от агентств, от которых он получает оповещения. Он также будет кэшировать последний момент времени, когда было принято оповещение от каждого агентства. Приемник маркирует агентства, которые, по его мнению, могут перестать существовать или являются неактивными. Он это делает, когда интервал неприслушивания агентства превышает период оповещения, установленный для агентства. Приемник может быть конфигурирован для ожидания нескольких периодов, прежде чем маркировать агентство как неактивное. Это позволит обрабатывать случаи пропущенных оповещений (ввиду, возможно, перегрузки в сети). Приемник будет обновлять список агентств в администраторе семантической среды каждый раз, когда он принимает оповещения.

Администратор семантической среды периодически спрашивает у приемника, имеются ли новые агенты. Администратор семантической среды проверяет список агентств и опрашивает каждое активное агентство, имеются ли у него новые агенты. Администратор семантической среды определяет этот запрос на основе времени создания последнего агента в агентстве, поддерживаемого локально, и текущего времени, определяемого по часам агентства. Агентство отвечает и также посылает новое значение времени создания последнего агента. Администратор семантической среды кэширует это значение в записи агентства. Если имеются новые агенты, то браузер информирует пользователя посредством диалогового окна и запрашивает у пользователя, не желает ли он просмотреть новых агентов.

Настоящее изобретение также поддерживает оповещения агентства с использованием процедуры однорангового обнаружения агента. В этой модели оповещения посылаются либо к серверу каталога и все клиенты проверяют его, либо непосредственно к клиентам по стандартному одноранговому протоколу публикации.

На фиг. 47-53 представлены примеры видов экрана, иллюстрирующие аспекты управления агентами со стороны KIS. Фиг. 47-50 иллюстрируют пример администратора управления агентством сервера интеграции знаний (KIS), показывая представления серверных агентов и серверных элементов. Фиг. 51 дополнительно иллюстрирует элементы пользовательского интерфейса администрирования для управления задачами SDG (просмотра для сбора информации), системными задачами (например, машины логического вывода), электронной почтой агентов системы (например, входящей почтой), календарем и контактами DSA и всеми таблицами данных SMS (объектов, семантических связей, категорий и т.д.). Фиг. 52 иллюстрирует пример диалога «свойства сервера» согласно настоящему изобретению в администраторе агентства KIS. Диалог иллюстрирует, каким образом администратор сервера может устанавливать свойства сервера, такие как имя сервера, имя дисплея, свойства хранилища данных SMS, свойства KDM (т.е. пути области знаний) и свойства DSA пользователя. Фиг. 53 иллюстрирует пример диалога «статистики сервера» в администраторе агентства KIS согласно настоящему изобретению. Диалог иллюстрирует отображение статистики, например, общего числа серверных агентов (стандартных агентов и элементов сопряжения), общего числа серверных стандартных агентов, общего числа серверных элементов сопряжения, общего числа представлений серверных агентов, общего числа подписок на серверных агентов, общего числа информационных объектов, сохраненных на сервере, общего числа семантических связей, общего числа пользователей на сервере (агентстве) и общего числа групп пользователей.

3. Сервер базы знаний (KBS).

Сервер базы знаний (KBS) является сервером, на котором находятся знания для KIS. В большинстве

приложений будет использоваться множество экземпляров KIS, но для заданной организации будет использоваться только небольшое число (или один) KBS. Это связано с тем, что KBS может повторно использоваться (они являются доменнозависимыми, но не зависят от данных). Например, фармацевтическая фирма может развернуть один KBS, инициализированный онтологией «Фармацевтика», но иметь несколько инсталляций KIS, вероятно, на каждое отделение сотрудников или на каждую группу сотрудников. KBS предпочтительно включает в себя следующие компоненты.

1. Одна или более онтологий, которые соответствуют одной или более семантических областей (областей знаний). На семантическую область ссылаются с использованием имени семантической области. Это имя, которое ссылается на путь области в семантической иерархии. Примерами являются «Промышленность.Технология», «Промышленность.Фармацевтика.Науки_о_Жизни», «Общее.Спорт.Баскетбол». Эти имена или пути могут быть определены глобально и уникальным образом (например, доменными именами Интернет), как обсуждено выше.

2. Одна или более таксономий, которые соответствуют поддерживаемым семантическим областям. Эти таксономии содержат иерархию имен категорий.

3. Механизм категоризации, который получает часть текста или XML и имя семантической области, с которой должна быть выполнена категоризация, и возвращает категории в той области, к которой принадлежит текст или XML, вместе с оценкой категоризации (по шкале 0-10 или предпочтительно 0-100).

4. XML-Web-сервис, который предоставляет API для добавления новых поддерживаемых семантических областей (и соответственно онтологий и таксономий), для перечисления категорий для заданной семантической области и для категоризации текста или XML-объекта данных.

5. Ссылка XML-Web-сервиса на другой, из которого данный KBS получает свои знания. В этом режиме KBS действует как посредник. KBS может быть инициализирован для того, чтобы действовать как посредник и получать свои поддерживаемые семантические области, онтологии и таксономии от других KBS.

Как прояснено выше, KIS (через KDM) периодически посылает XML-объекты к KBS для категоризации их для данной семантической области.

4. Информационный агент (платформа семантического браузера).

а. Обзор.

Клиент системы в предпочтительном варианте информационный агент настоящего изобретения включает в себя компоненты семантического браузера и пользовательский интерфейс, которые обеспечивают семантический опыт пользователя. В предпочтительном варианте информационный агент обеспечивает следующие услуги высокого уровня:

Обеспечивает пользователям возможности извлечения контекстно- и времязависимой информации посредством локальных и удаленных информационных агентов.

Обеспечивает возможность пользователям обнаруживать информацию на локальных и удаленных агентствах, которые представляются через агентов с помощью XML-Web-сервиса согласно настоящему изобретению. Эта информация предпочтительно классифицируется на хорошо известные семантические классы, такие как документы, электронная почта, списки распространения электронной почты, люди, события, мультимедиа, клиенты.

Позволяет пользователям просматривать семантическое представление информации, найденной через агентов согласно настоящему изобретению.

Позволяет пользователям публиковать информацию на агентстве.

Позволяет пользователям динамически связывать информацию на своем жестком диске, локальной сети или специальном агентстве, с информацией, найденной агентами из другого агентства. Это облегчает динамическое электронное связывание и управляемый пользователем просмотр ресурсов.

Преимущество информационного агентства настоящего изобретения заключается в том, что пользователи открывают агентов аналогично тому, как пользователи открывают документы из пространства имен их файловой системы. Информационный агент должен иметь свою собственную среду, которая открывает семантические «миры» информации. Например, компания ABC может иметь внутреннее агентство KIS, которое имеет агентов для внутренних документов, электронной почты и т.д. Кроме того, третьи стороны могут посещать агентства по сети Интернет для получения информации из сообщений промышленности, событий в промышленности и т.д. В предпочтительном варианте осуществления настоящего изобретения сотрудники компании ABC открывают агентов для обнаружения информации в Интернет, относящейся к их работе, а также для семантического связывания информации, являющейся внутренней для компании ABC, с информацией, которая является внешней, но релевантной для компании ABC.

б. Конфигурация клиента.

В предпочтительном варианте осуществления клиент системы способен семантически связывать информацию, найденную локально, а также на удаленных агентствах. Это предпочтительно выполняется посредством использования представленной семантической среды, содержащей агентства из глобального каталога агентств, агентства в локальной сети (публикуемые посредством системы многоадресной передачи или одноранговой публикации) и агентства из заказного каталога агентств, использующего обнару-

жение агентств. Предпочтительная конфигурация клиента основана на базовой структуре, имеющей агентов и локальные агентства, и включает в себя администратора семантической среды, который управляет локально сохраненными агентами и предпочтительными агентами, по существу, интегрируя метафоры (модельные представления) архива и предпочтений. Администратор семантической среды использует документы семантических запросов в семантической среде для представления знаний пользователям посредством браузера семантической среды. Конфигурация клиента также будет включать в себя информацию. Обнаружения агентств (например, список агентств, информацию каталога агентств и т.д.).

С. Спецификация базовой структуры клиента.

Обзор.

Спецификация базовой структуры клиента обеспечивает инфраструктуру сервиса для пользовательского интерфейса информационного агента и определяет базовые сервисы и интерфейсы, включает в себя базовые компоненты пользовательского интерфейса и обеспечивает наращиваемую, конфигурируемую среду для основных компонентов пользовательского интерфейса информационного агента. В этом разделе описана спецификация базовой структуры клиента в соответствии с предпочтительным вариантом осуществления настоящего изобретения. Ядро базовой структуры определяет базовые сервисы, конфигурацию, предпочтения и механизмы защиты. Базовые компоненты пользовательского интерфейса определяют сервисы пользовательского интерфейса и модули, которые поддерживают конфигурирование сервера и агента, управление и активизацию и в некоторой степени конфигурирование для базовой структуры семантического браузера. Базовые компоненты пользовательского интерфейса реализованы как расширение оболочки Windows и связанного пользовательского интерфейса (описано ниже). Базовая структура семантического браузера обеспечивает базовые сервисы управления запросом и результатами и базовую структуру для представления результатов. Специфические особенности пользовательского интерфейса, относящиеся к представлению семантического объекта, предпочтительно являются конфигурируемыми и наращиваемыми; даже поддержка устанавливаемого по умолчанию представления обеспечивается как предустановленное «расширение». Базовая структура семантического браузера предпочтительно реализуется как набор расширений поведения для существующих платформ, используемых в современной сети Web (например, Internet Explorer), и использует поддерживаемые функциональные средства XML, XSLT, HTML/CSS и DOM.

Контекст.

Базовая структура клиента строится на компонентах семантических сервисов настоящего изобретения, включая поддержку семантических запросов, контекстно- и времязависимую семантическую обработку и связывание информации и т.д. Базовая структура клиента предпочтительно строится как расширение оболочки и расширение платформы (например, Internet Explorer), что обеспечивает пользователям функциональные средства в контексте их существующих инструментальных средств и среды. Например, информационный агент может быть реализован как расширение оболочки (которое расширяет оболочку Windows и использует стандартные Explorer-представления и модели пользовательского интерфейса). В альтернативном варианте настоящее изобретение равным образом применимо в автономном приложении семантического браузера.

Требования.

Предпочтительными требованиями для базовой структуры клиента являются гибкость и наращиваемость. Это обеспечивает то, что пользовательский интерфейс может просто и легко адаптироваться, так как имеется много типов информационных объектов, профилей пользователей и т.д. Включены следующие требования:

- обеспечивать поддержку для поверхностей, чтобы управлять всем набором результатов запроса,
- обеспечивать возможность широкого диапазона подходов, включая списки, таблицы, временные слайды и т.д.,

- обеспечивать режим хранителя экрана (или эквивалентный ему режим),

- обеспечивать поддержку поверхностей, которые могут быть связаны с классом объекта,

- гарантировать наличие устанавливаемой по умолчанию поверхности, которая может обрабатывать все классы,

- поверхности должны быть настолько же просты, как и XSLT, но должны обеспечивать поддержку сценариев и, возможно, даже кодов (при соответствующих ограничениях секретности),

- обеспечивать поддержку просмотра ресурсов семантической среды в представлении результатов (для дополнения представления дерева агентов), включая агентов (интеллектуальных, неинтеллектуальных и специальных), агентства и элементы сопряжения,

- обеспечивать хорошо определенные интерфейсы между компонентами и гарантировать, что любая коммуникация может быть реализована через базовую структуру.

- обеспечивать надежную модель защиты во всей базовой структуре.

Ядро базовой структуры.

Администратор семантической среды (SEM) - управляет созданием, удалением, обновлением и просмотром агентов, элементов сопряжения и агентств на локальных машинах пользователей. Кроме того, SEM обеспечивает отслеживание многоадресных оповещений агентства, просмотр агентств по кли-

ентскому (пользовательскому) каталогу и просмотр агентств по глобальному каталогу агентства.

SEM включает в себя уровень хранения, который сохраняет метаданные каждого агента в системе, включая все атрибуты агентов, такие как имя агента, описание, время создания, время последнего использования, тип агента (интеллектуальный, неинтеллектуальный, специальный и т.д.), тип информационного объекта, представляемого агентом (для агентов, созданных на основе информационного типа), тип контекста, представляемого агентом, ссылку на XSLT или иной файл сценария, который представляет поверхность агента (включая предпочтения фильтрации/сортировки и другие схемы представления), информацию и метод уведомления (если запрашивается агентом) и буфер или путь файла/URL для SQML-запроса агента. Информационный агент (семантический браузер) может хранить эти метаданные агента в локальной базе данных, хранилище, подобном регистру Windows, или в хранилище XML-файлов на локальной файловой системе.

SEM также использует атрибут агента для указания, является ли агент предпочтительным агентом. Кроме того, SEM автоматически удаляет агентов, которые не являются предпочтительными агентами и возраст которых больше, чем конфигурируемый предельный возраст (например, две недели).

Расширение оболочки информационного агента и другие компоненты (такие как панель инструментов и диалог «открыть агента») используют SEM для обеспечения создания, удаления, просмотра, обновления агента и управления агентами посредством своего пользовательского интерфейса.

Администратор предпочтений. Этот компонент управляет всеми клиентскими предпочтениями, обеспечивая сервисы для поддержки предпочтений, осуществляет связь с серверами по мере необходимости для совместного использования предпочтений или поддержки перемещений и поддерживает установку и получение значений предпочтений от других компонентов. Этот компонент имеет связанный пользовательский интерфейс, а также некоторые более конкретные компоненты пользовательского интерфейса предпочтений. Предпочтения делятся на подкомпоненты и могут абстрагировать предпочтения для связанных классов клиентов. Они включают следующее.

Базовые предпочтения. Это включает в себя основные конфигурации, такие как профиль пользователя и персональную информацию.

Предпочтения поверхностей. Это связывает предпочтительные поверхности с классами объектов, а также предпочтительный список поверхностей и поверхности хранителя экрана. Могут иметься дополнительные установки связанных с поверхностью предпочтений. Этот компонент также управляет набором локально доступных поверхностей. Посредством этого компонента также предпочтительно управляются загружаемые поверхности.

Администратор уведомлений. Уведомления обеспечивают средство для указания пользователям, что имеется новая информация, доступная на конкретном интеллектуальном агенте. Пользователи дополнительно конфигурируют конкретного интеллектуального агента для поддержки или обеспечения уведомлений (по умолчанию это отключено для большинства интеллектуальных агентов), а также будут дополнительно конфигурировать то, каким образом представить уведомления пользователям. Эти уведомления представляются посредством компонента уведомлений пользовательского интерфейса.

Администратор уведомлений обеспечивает управление фоновой работой опрашивания о запросах для соответствующего набора интеллектуальных агентов. Администратор динамической информации является параллельным компонентом, который обеспечивает подобные сервисы для браузера результатов.

Администратор уведомлений собирает список интеллектуальных агентов, маркированных для уведомлений и периодически опрашивает связанные серверы о наличии новой информации. «Новое» здесь определяется как «с времени последнего опроса (или запроса)». Каждый раз, когда на этот опрос дается ответ, он включает в себя индикатор временной метки, который администратор уведомлений должен сохранить вместе с агентом.

Пользовательский интерфейс, связанный с конфигурированием администратора уведомлений, предпочтительно реализуется в координации с представлением дерева агентов. Это позволяет осуществить уведомления (например, опцию всплывающего меню «Уведомить» каждого интеллектуального агента). Администратор уведомлений может также поддерживать альтернативы для уведомления пользователя, когда имеются доступные новые результаты. Некоторые опции включают в себя стили отображения (например, жирным шрифтом, в цвете и т.д.) для агента в представлении дерева агентов, диалог напоминания, аудиоуведомление или более необычные действия, такие как IM- или SMS-уведомления.

Клиентская защита. Вопросы клиентской защиты относятся к коду расширения и поверхностям. Поверхности предпочтительно соответствуют XSLT, но могут также поддерживать сценарии. Кроме того, генерируемые HTML могут включать в себя ссылку на компоненты и режимы соответственно ActiveX. «Песочница» (механизм обеспечения защиты, предусматривающий изоляцию на время выполнения загружаемого кода в ограниченную среду) представлений может включать в себя ограничения защиты, которые запрещают поверхности исполнять потенциально злоумышленный код посредством сценариев. Например, реализация может полностью запретить любой неподписанный код (включая режимы ActiveX и DHTML).

Все коммуникации клиент-сервер с агентствами предпочтительно скрыты от опубликованных ин-

терфейсов (для поверхностей), которые третьи стороны будут настраивать для получения заказных (настроенных) поверхностей. Путем изолирования функциональных средств извне основного клиентского времени исполнения, риск ухудшения защиты может быть снижен.

Базовые компоненты пользовательского интерфейса.

Представление дерева агентов. Это представляет собой представление дерева в расширении оболочки, которое поддерживает многие базовые компоненты пользовательского интерфейса для управления и активизации агентов.

Пользовательский интерфейс для просмотра ресурсов семантической среды. Это обеспечивает пользовательский интерфейс для обеспечения возможности пользователю просматривать ресурсы семантической сети. Примером этого является диалог «Открыть агента». Это дополняет представление дерева агентов, которое также отображает иерархическое представление пространства имен (см. виды экранов).

Инспектор агентов. Это обеспечивает пользовательский интерфейс для просмотра свойств или редактирования (в случае созданных пользователями интеллектуальных агентов) индивидуального агента, элемент сопряжения или агентства.

Хост браузера. Это предпочтительно «упаковщик» на ядре семантического браузера (например, рабочий цикл браузера Internet Explorer), который обеспечивает возможность презентации пользовательского представления агентов, агентств и элементов сопряжения в представлении дерева агентов. Он предпочтительно не имеет какого-либо собственно пользовательского интерфейса и является сопрягающим компонентом между расширением оболочки и базовой структурой браузера. Этот компонент также предпочтительно несет ответственность за координацию некоторых функциональных средств браузера с пользовательским интерфейсом оболочки Windows, включая, в частности, механизм навигации («назад/вперед») для обеспечения непрерывного пользовательского опыта в направлениях назад/вперед (где пользователю необходимо иметь дело только с одним архивным списком «назад/вперед»).

Базовые предпочтения UI. Это обеспечивает пользовательский интерфейс, связанный с семантической средой, сервером, персоной и управлением агентом, а также другими разнородными настройками предпочтений. Это предпочтительно включает табличное диалоговое взаимодействие со свойствами примитивов, предпочтительно разделенными на несколько таблиц по функциональным областям. В предпочтительном варианте осуществления это будет пользовательский интерфейс, реализующий диалоговое окно с выбором клавишей табуляции.

Предпочтения поверхности UI. Это обеспечивает пользовательский интерфейс для предпочтений, связанных с управлением поверхностью. Это предпочтительно является диалоговым взаимодействием с таблицей свойств. Список доступных поверхностей должен быть представлен как список для выбора. Пользовательский интерфейс позволяет пользователям устанавливать текущие поверхности как отличающиеся от устанавливаемых по умолчанию поверхностей. Это также позволяет пользователям делать их текущие поверхности устанавливаемыми по умолчанию. Для предпочтений относительно поверхностей, устанавливаемых на основе по каждому агенту, это предпочтительно позволяет пользователям выбирать поверхность для текущего выбранного или открытого агента.

Уведомление UI. Пользовательский интерфейс, связанный с конфигурированием администратора уведомлений, предпочтительно реализуется координированным образом с представлением дерева агентов. Администратор уведомлений может также поддерживать альтернативы для уведомления пользователей, когда имеются новые доступные результаты. Некоторые опции включают в себя стили отображения (например, жирным шрифтом, в цвете и т.д.) для агента в представлении дерева агентов, диалог напоминания, аудиоуведомление или более необычные действия, такие как IM-или SMS-уведомления. В предпочтительном варианте осуществления пользовательский интерфейс должен включать в себя диалоговое окно с выбором клавишей табуляции (или эквивалентное решение), чтобы позволить пользователям выбирать из вышеупомянутых схем уведомления (и тому подобного).

Хранитель экрана. Пользовательский интерфейс предпочтительно предусматривает специальную модальность для браузера результатов, функционирующую подобно хранителю экрана, заполняя экран «сценическим» изображением. В предпочтительном варианте осуществления для режима хранителя экрана должны использоваться специальные поверхности. Эти поверхности могут акцентировать динамическое отображение, которое может использовать большую площадь экрана, но могут также использовать более крупные шрифты и более широкий контур.

Базовая структура браузера.

Браузер результатов. Браузер результатов обеспечивает отображение результатов запросов и информацию на любом открытом локальном ресурсе. Браузер результатов предпочтительно получает один или более XML-файлов от администратора запросов и объединяет их единый XML-файл, который представляет список объектов. Список объектов может фильтроваться или сортироваться как первоначальный этап обработки.

Список как структура преобразуется посредством специального класса поверхности (таблица XSLT-преобразования, возможно включающая в себя некоторый сценарий), который обрабатывает списки. Поверхность списка создает первичную DHTML-структуру (или подобную ей структуру), например, список, таблицу или, возможно, синхронизированную последовательность. Поверхности объекта управ-

ляют индивидуальными DHTML-элементами, которые представляют информацию для каждого экземпляра объекта. Поверхности списка могут управлять координацией поверхностей индивидуальных объектов (отображающих класс объекта на поверхность), но браузер результатов для простоты предпочтительно обеспечивает отображение по умолчанию класса на поверхность.

Пользователи могут предпочитать некоторую форму презентации и могут выбирать установленные по умолчанию поверхности (как для списка, так и для классов объектов). Исходный запрос (например, SQML) может также включать в себя параметры, которые указывают, какие поверхности должны быть использованы (особенно какая поверхность списка). Они будут введены в браузер результатов вместе с результатами. Браузер результатов использует средства администратора поверхности для выбора корректной поверхности для применения. Могут использоваться различные правила, касающиеся того, каким образом пользовательские предпочтения и предпочтения агента (автора) комбинируются и приоритизируются.

Если результаты запроса составлены из множества отдельных файлов, браузер результатов должен объединить их в единый XML-документ для обеспечения непрерывного пользовательского опыта. Предпочтительный вариант осуществления предусматривает дополнительную динамическую обработку результатов. Этот динамический режим обновления предпочтительно реализуется с использованием различных шаблонов или, возможно, метода сценария в XSLT-шаблоне. Альтернативно, поверхности списка могут потребовать определенного режима (или локального компонента рабочего цикла) для управления логикой дополнения документа без искажения пользовательского контекста.

Администратор запросов (клиентский процессор семантических запросов). Администратор запросов обеспечивает обработку информационного обмена с сервером(ами), выполняющими запросы на информацию и собирающими XML-результаты. Полученный в результате XML направляется в браузер результатов для презентации пользователям.

Администратор запросов предпочтительно обеспечивает сервисы для поддержки функциональных возможностей интеллектуальной лупы. Когда делается запрос интеллектуальной лупы, результаты возвращаются как XML и посылаются в браузер результатов, предпочтительно маркированными для указания, что они являются результатами интеллектуальной лупы для данного объекта. Администратор запросов предпочтительно включает следующие подкомпоненты, которые обеспечивают индивидуальные сервисы для выполнения требований запроса.

SQML-интерпретатор. Этот компонент должен разлагать пришедший SQML на набор запросов, возможно, со связанными ресурсами. Каждый запрос или связь ресурсов разрешается (отображается) на ресурс с соответствующим протоколом (например, HTTP или один из ряда локальных псевдопротоколов, подобных outlook: или document:), и диспетчеризуются в соответствующий обработчик протокола. Данный SQML-файл может представлять собой смесь сетевых и локальных типов ресурсов.

Администратор обработчика ресурсов. Это представляет собой центральный механизм регистрации для обработчика ресурсов. Это минимальный уровень, который связывает протоколы и псевдопротоколы с обработчиками и упрощает диспетчеризацию запросов ресурсов.

Обработчики ресурсов. Это компоненты, которые инкапсулируют специфические особенности доступа к ресурсам с данного «сервера». Обработчик ресурсов не разрешает никаких связанных ресурсов. Это предпочтительно является обязанностью SQML-интерпретатора (т.е. SQML-интерпретатор должен уже иметь разрешенные связанные ресурсы и обеспечивать связанные метаданные как часть запроса ресурса к этому обработчику). Если ресурсом является сервис семантической сети Web, то данный компонент предпочтительно связывает пакет и выдает его посредством HTTP. Если ресурс является локальным ресурсом (например, ресурсом протоколов document: или Outlook:), то обработчик ресурсов обрабатывает ресурс непосредственно. Для документов, обработчик ресурсов направляет документ (файл: URL) на процессор выделения семантического смысла, резюмирования и категоризации для выделения метаданных. Для электронной почты обработчик ресурсов выделяет сообщения с сервера электронной почты (exchange) или из локальных .PST-файлов. Заметим, что если имеются связи к локальному ресурсу, обработчик локальных ресурсов должен выполнять обработку, которая фильтрует результаты для семантической связанности понятий. Это может быть настраиваемым для обработчика в целях достижения большей эффективности, но центральный обобщенный процессор связанности должен обеспечивать сервисы для большинства случаев.

Процессор связанности. Это обеспечивает пункт сбора логики для сравнения объектов на связанность. Сравнение предпочтительно зависит от совокупности используемых схем, но в остальном представляет собой простую операцию - при условии наличия двух объектов дать меру их связанности.

Администратор фильтрации/сортировки. Администратор фильтрации/сортировки поддерживает применение фильтрации и сортировки к спискам результатов, обеспечиваемых браузером результатов. Администратор фильтрации/сортировки использует сервисы компонента предпочтений фильтрации/сортировки для получения пользовательских предпочтений для текущих настроек. Главная функция этого компонента состоит в разрешении основных предпочтений, предпочтений на основе по каждому агенту, и любых установок, определенных в действительных результатах (это может поддерживаться или нет). Этот компонент уведомляется посредством компонента предпочтений фильтрации/сортировки, ко-

гда пользователи изменяют текущие применяемые фильтры и средства сортировки. Поскольку связанный пользовательский интерфейс является частью панели инструментов, связанной с расширением оболочки (т.е. его правая панель «Представление»), но применение функций осуществляется в пространстве браузера результатов, то управление в типовом случае является косвенным.

Режим лупы. Когда активизируется интеллектуальная лупа, браузер результатов должен генерировать требования (запросы) лупы на объекты, которые выбраны пользователем. Запросы являются асинхронными, так что пользователи могут выбирать запросы интеллектуальной лупы в отношении различных объектов и просматривать результаты по мере того, как они возвращаются. Предложенный для этой цели пользовательский интерфейс связан с резервированием некоторой функциональности для пиктограммы интеллектуальной лупы. Когда пользователь находится в режиме интеллектуальной лупы и выполняет щелчок на пиктограмме интеллектуальной лупы, выдается запрос, и пиктограмма изменяет свое состояние, показывая, что запрос обрабатывается. Когда результаты возвращаются, они обрабатываются браузером результатов и специализированными шаблонами интеллектуальной лупы в поверхностях и пиктограммой интеллектуальной лупы для изменений объекта, чтобы показать, что результаты доступны. Повторный щелчок на пиктограмме отобразит результаты интеллектуальной лупы конкретным образом, соответственно поверхности (см. пример пользовательского интерфейса панели интеллектуальной лупы). Если запрос возвращается довольно быстро, то вся функция воспринимается как всплывающее меню, активизируемое щелчком.

Представление глубинной информации. Если глубинная информация не доступна в исходных результатах, то этот компонент генерирует связанный запрос. Запрос предпочтительно является асинхронным. Когда результаты возвращаются в браузер результатов, они обрабатываются посредством соответствующей поверхности (с использованием специального шаблона глубинной информации для каждой поверхности), и результирующие HTML-данные вводятся в полученный в результате документ под ассоциированным объектом. Основная поверхность для схемы вводит элемент глубинной информации в HTML-данные для объекта, так что браузеру результатов известно, куда ввести результаты. Когда глубинная информация доступна (либо как часть исходных результатов, либо в ответ на запрос глубинной информации), поверхность либо отображает ее непосредственно, либо указывает, что она присутствует, и некоторый определяемый поверхностью пользовательский интерфейс позволит пользователям осуществить отображение (например, как всплывающее окно).

Администратор контекстной информации. Для объектов, отображаемых текущим образом в браузере результатов, некоторые уведомления предпочтительно обеспечиваются по умолчанию. Пользователям будут предоставляться два класса новой или дополнительной информации.

1. Дополнительные результаты, которые были добавлены к серверу с момента, когда пользователь осуществил исходный запрос. Это особенно полезно для таких элементов, как заголовки или активные темы в электронной почте. Результаты обрабатываются браузером результатов путем вставки новых объектов в представление.

2. Шаблоны контекстов и связанная информация, которые будут представлять интерес для пользователя. Это генерируется дополнительными запросами к специфическому агенту (интеллектуальному агенту, специальному агенту, элементу сопряжения или агентству) с использованием конкретного объекта в качестве контекста. Результаты обрабатываются подобно тому, как обрабатываются результаты представления глубинной информации и режима интеллектуальной лупы, путем обработки XML-данных, возвращенных от запроса, и вставки результирующих HTML-данных в существующие HTML-данные для объекта. Поверхность управляет механизмами отображения и пользовательскими интерфейсами. Примером связанной информации являются «Новости, вызывающие прерывание», ассоциированные с объектом.

Администратор поверхности - поддерживает пользовательские предпочтения для поверхностей списка, поверхностей объекта и зависимости между поверхностями списка и объекта (некоторые поверхности объекта могут иметь смысл только для данной поверхности списка). Администратор поверхности также поддерживает параметры для каждой поверхности, которые указывают ограничения для поверхности, то есть какое количество ресурсов экрана она требует, или модальности (режимы), к которым она наилучшим образом применима. Значительный уровень интеллекта (встроенные вычислительные средства) способствуют выбору браузером результатов поверхностей для заданного диапазона ограничений на экран и размер окна, а также модальностей, доступность, язык и другие ограничения. Первоначальные версии, вероятно, будут намного проще.

Шаблоны поверхностей. Это описывает структуру поверхности и то, откуда она применяется в браузере результатов. Поверхность предпочтительно является XSLT-шаблонами, которые преобразуют XML результатов в XHTML (и/или другие языки, подобные SVG) или специализированные платформы представлений, подобные Flash MX и ActionScript. Шаблоны также могут вставлять информацию стиля (оформления), например, для CSS-оформления. Результирующий код представления (например, XHTML) может ограничить включение кода по причине защиты. Базовый код в браузере результатов активизирует поверхности. Предпочтительный вариант осуществления включает в себя следующие классы поверхностей.

Поверхности списка (или поверхности топологии). Поверхности списка используются для преобразования списка объектов, возвращенных из запроса, в некоторую общую структуру представления. Это может быть простой список, таблица, временная последовательность слайдов. Поверхности списка не являются специфическими для схемы или объекта, хотя они могут поддерживать только некоторые поверхности, которые могут работать в пределах ограничений, которые определяет ассоциированная форма представления. Например, поверхность списка, которая определяет структуру таблицы, может требовать или предпочитать поверхности объекта, которые могут сформировать информацию в мелком прямоугольном формате.

Поверхности объекта. Поверхности объекта являются конкретной схемой и генерируют представление для индивидуального объекта для заданного типа информационного объекта (или информационного класса). Можно определить поверхность для обобщенного супер-класса (или любого другого супер-класса), который может служить как установленная по умолчанию поверхность для диапазона производных классов или подклассов (предположительно путем пропуска некоторых деталей).

Поверхности контекста. Поверхности контекста связаны с конкретным шаблоном контекста и генерируют представление, которое будет наиболее эффективно отражать контекст, указанный шаблоном.

Поверхности элемента сопряжения. Поверхности элемента сопряжения предназначены для представления результатов с элементов сопряжения. Эти поверхности должны позволять пользователям просматривать результаты посредством агентов, содержащихся в элементах сопряжения, посредством типа информационного объекта или посредством объединенного представления, которое отображает все результаты, как если бы они происходили из одного и того же источника.

Поверхности предпочтительно моделируют ограничения, такие как модальность и область отображения представления путем обработки ограничений (вводимых как параметры статически или динамически посредством событий в самом ядре браузера). Это предпочтительно поддерживается путем наложения ограничения, что поверхности списка должны определять только допустимые поверхности объекта. В альтернативном методе поверхности объекта могут проектироваться для заданной поверхности списка, и браузер результатов/администратор поверхности выбирает поверхности объекта для текущей поверхности списка.

Детали поверхности списка. Пользователи могут выбирать отдельную поверхность списка для текущего представления и задавать ее как устанавливаемую по умолчанию. Поверхности списка могут также быть связаны с индивидуальными агентами, причем в этом случае общий оператор умолчания перепределяется. Браузер результатов активизирует поверхность списка для обработки списка результатов, хотя поверхность списка предпочтительно реально не обрабатывает индивидуальные объекты. Она создает некоторый экземпляр объекта в базовом представлении (например, временной ввод (запись) в последовательности, ячейку в таблице или элемент в списке), и затем поверхности объекта заполняют детали.

Детали поверхности объекта. Поверхности объекта преобразуют конкретную схему в XHTML. Поддержка асинхронных результатов запроса для таких объектов, как глубинная информация и информация шаблона контекста, обеспечивается путем активизации ассоциированных шаблонов из браузера результатов (через DOM) по XML результатов запроса, и затем ввода полученных в результате XHTML-данных в результирующий документ через интерфейсы DOM. Предпочтительно имеется несколько отдельных шаблонов в поверхности объекта, включая

шаблон главной схемы - это главный элемент, который генерирует XHTML для отображения по умолчанию. Это должно создавать упаковщиков (системной оболочки для стандартизации внешних обращений) для глубинной информации, информации интеллектуальной лупы, информационного содержания шаблона контекста и любого сценария, который обеспечивает пользовательское управление над ассоциированным отображением;

шаблон глубинной информации - этот шаблон обрабатывает метаинформацию для глубинной информации. Он может быть вызван для получения вставленной глубинной информации, предоставляемой с исходными результатами, или может быть вызван для обработки асинхронно запрашиваемой глубинной информации. В любом случае, он предпочтительно генерирует XHTML в некоторой форме, которая вводится под элементом упаковщика для глубинной информации.

Вставка, вероятно, осуществляется в XSLT для встроенной глубинной информации и осуществляется посредством DOM-вставки для результатов запросов глубинной информации.

Шаблон контекстной информации - этот шаблон обрабатывает результирующую информацию для результатов информационных запросов. Он генерирует XHTML-данные в некоторой форме, которые вводятся под элементом упаковщика в реальную (динамическую) информацию. Этот ввод осуществляется через DOM-вставку для результатов запроса глубинной информации.

Шаблон информации интеллектуальной лупы - этот шаблон обрабатывает результирующую информацию для результатов интеллектуальной лупы. Он генерирует XHTML-данные в некоторой форме, которые вводятся под элементом упаковщика в реальную информацию. Этот ввод осуществляется через DOM-вставку для результатов запроса глубинной информации. В предпочтительном варианте осуществления шаблон не может модифицировать другие компоненты XHTML (даже для того же самого объек-

та), так что он будет вплоть до браузера результатов координировать изменения пользовательского интерфейса, которые показывают когда становятся доступными результаты глубинной информации, реальной информации или интеллектуальной лупы. Базовая структура требует использования определенных пиктограмм (также для согласованности) и чтобы они имели регулярные имена или типы элементов, что позволит браузеру результатов находить их и модифицировать по мере необходимости. Кроме того, браузер результатов может создавать и представлять объекты для указания изменений результатов. Генерируемый шаблоном сценарий может реагировать на эти события и отображать ассоциированную информацию, как это требуется.

Поверхности, установленные по умолчанию. В предпочтительном варианте осуществления предусматривается набор поверхностей, установленных по умолчанию. Это предпочтительно включает в себя поверхности для базовых классов объектов и малого набора поверхностей списка, которые обеспечивают возможность реализации разнообразных представлений результатов запросов. Предпочтительные поверхности списка включают в себя следующее:

- отображение детального списка (подобно представлению деталей Windows Explorer),
 - табличное представление пиктограмм (вновь подобно представлению пиктограмм Windows Explorer, но в несколько более расширенном виде),
 - временное представление презентации.
- e. Клиентская базовая структура.

В предпочтительном варианте осуществления клиент системы включает в себя расширения оболочки презентатора и поверхности, используемые презентатором для отображения информации с контекстом и смыслом.

Расширение оболочки. Расширение оболочки программы Explorer является компонентом программного обеспечения Microsoft Windows, который расширяет оболочку Windows заказным (специализированным) кодом. Расширения оболочки позволяют приложениям использовать оболочку как заказного клиента, а также обеспечивают сервисы, такие как чистая интеграция с рабочим столом, файловой системой, Internet Explorer и т.д. Примеры установленных по умолчанию расширений оболочки включают «Мои документы», «Мой компьютер», «Мои сетевые узлы», «Буфер повторного использования», «Internet Explorer». Использование расширения оболочки в предпочтительном варианте осуществления настоящего изобретения имеет следующие преимущества.

1. Оно обеспечивает очень четкий способ обеспечения пользовательского опыта, позволяющего непрерывно интегрироваться с тем, каким образом специалисты в области информационных технологий в настоящее время просматривают информацию. В свою очередь, это исключает необходимость в разработке специализированных клиентов и обеспечивает нестандартную интеграцию с Microsoft Internet Explorer, «Моими документами» и т.д.

2. Оно включает в себя современную сеть Web и обеспечивает путь для перемещения содержимого современной сети Web в информационную нервную систему согласно настоящему изобретению. Например, пользователи предпочтительно могут перемещать и оставлять документы со своего жесткого диска (посредством Microsoft Explorer) или из Интернет (посредством Internet Explorer) в удаленные агенты в расширении оболочки настоящего изобретения. Это затруднительно и не выводится очевидным образом с использованием специализированного клиента. Тем не менее настоящее изобретение предусматривает переносимость (программы) на специализированного клиента или на эквивалент расширения оболочки или на отличную от Windows операционную систему и операционные системы непersonальных компьютерных устройств. Расширения оболочки настоящего изобретения обеспечивают представление пользовательской семантической среды (например, архива, предпочтений и других представлений). В предпочтительном варианте осуществления расширение оболочки обеспечивает следующее.

1. Позволяет пользователям открывать агента, документ, папку или адрес в семантической среде семантического браузера. В случае агента клиент отображает настроенное (заказное) диалоговое окно «Открыть агента», которое позволяет пользователям просматривать ресурсы семантической среды семантического браузера. Это предпочтительно включает агента в пользовательский список «Мои агенты», агентства в глобальном каталоге агентств, агентства в локальной сети (оповещаемые посредством многоадресной передачи) и агентства в любом специализированном каталоге агентств, которые конфигурированы пользователями. Опция «Вставить релевантное представление экрана на пользовательский интерфейс», открывающая агента, приводит в результате к клиентскому отображению результатов запроса данного агента. Открытие документа открывает XML-метаданные для данного документа, согласованного со схемой для типа объекта документа. Открытие папки открывает XML-метаданные для папки файловой системы. Пользователи имеют возможность открыть немедленное или глубинное содержимое папки посредством самой папки. Открытие адреса позволяет пользователям войти в любой адрес, открываемый клиентской базовой структурой. Это включает URL (который открывает XML-метаданные для документа), документы файловой системы, агентов или объекты (см. ниже «Соглашения об именовании URL»). В случае агентов URL агента предпочтительно вводится следующим образом: Агент://<Имя Агента>@<Имя Агентства><имя домена>. Это аналогично соглашению об именовании для URL HTTP вида: http://<URL>. Префикс Агент://требуется в этом случае, так как опция «Открыть адрес» может от-

крыть любой адрес. В случае опции «Открыть агента» пользователям не требуется добавлять префикс; клиентская базовая структура автоматически определяет URL для включения префикса. Это аналогично тому, как пользователи могут ввести www.foo.com в современный браузер без определения префикса http://. Ожидается, что клиент позволит пользователям открывать другие объекты, например, файлы Microsoft Outlook.PST.

2. Позволяет пользователям просматривать ресурсы, осуществлять подписку и отменять подписку на агентов в данном агентстве, которое поддерживает пользовательское состояние.

3. Позволяет пользователям сохранять активизированных агентов или результаты семантических запросов в списке «Мои агенты».

4. Позволяет пользователям создавать элементы сопряжения и добавлять и удалять агентов в элементах сопряжения (включая использование метода «переместить и оставить»).

5. Уведомляет пользователей, если со времени последней проверки появились новые агентства на каком-либо из каталогов агентств (например, глобальном каталоге агентств, локальной многоадресной сети или каком-либо из специализированных каталогов агентств).

6. Уведомляет пользователей, если со времени последней проверки появились новые агенты на каком-либо из агентств.

7. Обеспечивает доступ с использованием метода «переместить и оставить» к реляционным семантическим запросам объектов в семантической сети. Расширение оболочки позволяет пользователям перемещать и оставлять документ из семантической сети (на локальном дисковом, в сетевой окрестности, в интранет или Интернет) в папке оболочки, представляющей агента. Это запускает удаленную процедуру вызова XML-Web-сервиса для данного агентства с метаданными документа в качестве аргумента.

8. Обеспечивает доступ «вставки» к объектам, скопированным в системный буфер обмена. Настоящее изобретение использует системный буфер обмена, чтобы обеспечить возможность пользователям копировать любой объект для будущего доступа. Кроме того, буфер обмена обеспечивает возможность пользователям копировать любой объект из других приложений, например, из приложений Microsoft Office (например, объектов электронной почты из Outlook), из мультимедийных приложений и копировать данные из любого приложения.

9. Позволяет выбирать агента в качестве интеллектуальной лупы. Интеллектуальная лупа позволяет пользователям просматривать объекты в представлении результатов на основе контекста из агента или любого объекта, который может быть скопирован в системный буфер обмена. Например, если объект документа находится в представлении результатов и пользователь указывает на связь, представляющую объект, то отображаются метаданные объекта. Если, однако, выбирается интеллектуальная лупа (например, путем вставки ее в таблицу результатов), и пользователь указывает на объект, то отображается информация, относящаяся к этому объекту в интеллектуальной лупе, и объект под курсором. Например, если пользователи копируют "People.Research.All" («Люди.Исследования.Все») в буфер обмена и вставляют это как интеллектуальную лупу, то указание документа приведет к отображению метаданных во всплывающем контуре в следующем виде: «Найти людей в "People.Research.All"», которые являются экспертами по данному документу». Другими примерами могут служить следующие: «Найти 3 людей, которые, возможно, написали этот документ» и «Найти 78 сообщений электронной почты, относящихся к этому документу, посланных людьми из "People.Research.All"». Пользователи принимают решение, следует ли активизировать любую из связей в метаданных во всплывающем контуре. В альтернативном варианте контур может отображаться в боковом меню и не требует контура. Если интеллектуальная лупа вставляется в буфер обмена, то расширение оболочки предпочтительно осуществляет информационный обмен с системой и изменяет курсор мыши для отражения имени выбранного агента. Интеллектуальная лупа предпочтительно имеет глобальную оценку, поскольку она копируется из буфера обмена. Иными словами, например, все версии Windows Explorer и Internet Explorer «видят» интеллектуальную лупу и реализуют на ее действия. В предпочтительном варианте имеется инструментальное средство «Интеллектуальная лупа» в панели инструментов информационного агента, которое применяется к текущему объекту в буфере обмена (например, агенту или иному объекту). По умолчанию выбор инструментального средства «интеллектуальная лупа» будет отменяться после щелчка на связи в системе. Пользователи могут «закрепить» интеллектуальную лупу. Если интеллектуальная лупа «закреплена», то интеллектуальная лупа остается активной до тех пор пока пользователь в явном виде не отменит ее выбор. В предпочтительном варианте для закрепления интеллектуальной лупы пользователи выбирают опцию «Вставить как интеллектуальную лупу и закрепить» на панели инструментов.

10. Позволяет пользователям «вырезать» результаты агента из расширения оболочки и отобразить их в пристыкованном (плавающем) представлении на рабочем столе. В этом представлении окно браузера результатов агента действует как семантический тиккер. Эта функция позволяет пользователям непрерывно отображать семантическую информацию при продолжении другой работы.

11. Позволяет пользователям запускать агента для использования в качестве хранителя экрана.

12. Позволяет пользователям просматривать и активизировать доступные поверхности в глобальном каталоге агентств.

Презентатор. Презентатор является набором локальных компонентов (например, встроенных моду-

лей браузера), которые осуществляют семантические запросы из сценариев (или других встроенных модулей) и передают их в XML-Web-сервис агентства сервера интеграции знаний (KIS). Настоящее изобретение обеспечивает преобразование результатов семантических запросов и пропускает XML в другие режимы или сценарии для представления в конечном счете пользователям. В предпочтительном варианте презентатор вызывается расширением оболочки в файле SQML. Система предпочтительно осуществляет связь с XML-Web-сервисом агентства непосредственно. Система разрешает SQML-файл и активизирует вызовы для открытия XML-информации из местного или удаленного источника (через XML-Web-сервисы агентств, на которые есть ссылки в SQML-файле). Альтернативно, если URL агента введен в систему, презентатор непосредственно открывает URL путем активизации его посредством вызова в XML-Web-сервис агентства, на котором находится агент. В предпочтительном варианте система вызывает соответствующий метод для соответствующего типа семантического объекта. Примерами устанавливаемых по умолчанию типов объекта являются SEMANTICOBJECTYPEID_EVENT, SEMANTICOBJECTYPEID_EMAILMESSAGE, которые определены в файле заголовка (semanticruntime.h). Предпочтительный вариант осуществления допускает регистрацию новых типов семантических объектов через интерфейс программирования приложений (API) RegisterSemanticObjectType. Процессор семантического запроса в агентстве возвращает соответствующие XML-результаты с использованием типа семантического объекта в качестве фильтра.

В предпочтительном варианте поверхность согласно настоящему изобретению использует (см. ниже) XSLT (и/или сценарий) для преобразования XML, возвращаемого из базовой структуры (из XML-Web-сервиса) в DHTML. Расширение оболочки позволяет пользователям выбрать новую поверхность для текущего запроса. Поверхности предпочтительно являются специфическими для типа объекта, шаблона контекста (для специальных агентов), или элемента сопряжения (для элементов сопряжения). Поверхности могут также специализироваться на основе имени/пути семантической области или онтологии агента и на основе других атрибутов, таких как пользовательская персона, состояние, местоположение и т.д. Каждый агент конфигурируется на агентстве с устанавливаемой по умолчанию поверхностью. Настоящее изобретение также может использовать специализированные поверхности, данные которых могут публиковаться на корневом каталоге агентства (например, на глобальном каталоге агентств).

Клиент предпочтительно загружает поверхность либо с агентства для объявленного агента или с центрального сервера (например, с глобального каталога агентств) и применяет ее для текущего представления. Клиент дополнительно может включать пользовательские предпочтения, игнорируя поверхности агента или согласуя их в части пользовательского интерфейса.

Помимо типа поверхности (например, поверхность объекта, поверхность списка/топологии, контекстная поверхность, поверхность элемента сопряжения и т.д.) в предпочтительном варианте поверхности разделяются на следующие категории:

- поверхности шаблона проектирования,
- поверхности шаблона цвета,
- поверхности шаблона анимации.

Семантические поверхности предпочтительно должны быть интерактивными, за исключением случаев, когда они отображаются как части «вырезок» (см. выше) или как хранитель экрана. Каждая поверхность позволяет пользователю выбирать конкретный пункт в «семантическом представлении». Например, если поверхность первоначально отображает только первые 25 элементов, поверхность должна иметь соответствующую подсветку выбранной позиции (или иной механизм пользовательского интерфейса), чтобы позволить пользователю просмотреть следующие 25 элементов, прокрутить их вперед, назад и т.д. Некоторые поверхности имеют опцию «Режим реального времени». В этом режиме поверхность непрерывно извлекает новые объекты из XML-Web-сервиса путем опроса. Поверхности обеспечивают опрос XML-Web-сервиса для получения новой информации на основе схемы желательных объектов. В предпочтительном варианте отсутствуют уведомления клиента, поскольку агентство не поддерживает какого-либо специфического для клиента состояния по причинам масштабируемости.

Поверхности предпочтительно включают в себя режим реального времени. Эти поверхности должны быть интеллектуальными в том, что они должны циклически изменяться по всем объектам (т.е. представлять, упорядочивать или подсвечивать) на основе приоритета. Например, если презентатор транслирует информацию, указывающую, что новый объект помещен на агентство, поверхность немедленно отображает/переупорядочивает/высвечивает это и продолжает представление остальных объектов. Презентатор определяет порядок и операции поверхности с динамикой, определяемой различными установками сортировки и фильтрации. Это создает восприятие семантического представления как происходящее в реальном времени. В предпочтительном варианте это происходит, если имеются новые данные, к которым пользователь имеет доступ с использованием поверхностей. Если список сортируется по времени, то представление в режиме реального времени может ввести пользователей в заблуждение вследствие перехода пользовательского интерфейса в интерактивный режим. Опция пользовательских предпочтений в некоторых режимах (например, в режиме хранителя экрана) автоматически устанавливает поверхность на отражение новых данных (например, путем прокрутки вверх отсортированного списка, когда новые данные вставляются наверх списка).

В альтернативном варианте поверхности проектируются для специализации их представления на основе ряда доступных окон представления. Например, поверхность может изменяться от статического режима до динамического режима путем отображения информации с использованием постепенного введения изображения и выведения изображения, если, например, окно представления относительно мало. Поверхности являются предпочтительно модальными в зависимости от ожидаемого уровня пользовательского взаимодействия. Например, хранитель экрана действует отличающимся образом из браузера; плавающее представление также отличается (не только тем, что оно меньше, но и ввиду того, что оно является некоторого рода фоновым представлением, а не фокусом пользовательского взаимодействия). Если представление минимизировано или скрыто, может использоваться альтернативный режим (особенно для указания наличия новой информации). Примерами являются аудиоуведомление, предупреждения подобные напоминаниям, показ и мерцание начальной строки (полосы) (подобно напоминаниям в Outlook). Агенты могут использоваться для отправки электронной почты, телефонных уведомлений и уведомлений мгновенного сервиса сообщений. В альтернативном варианте настоящее изобретение предусматривает агента, который посылает сообщения на Web-сайт (например, автоматическая генерация HTML-содержимого для календарей событий).

Альтернативно, поверхности могут генерировать аудио-визуальную информацию. Например, текстово-речевая поверхность может считывать объект электронной почты. Эта функция имеет большое потенциальное значение для недееспособных пользователей или для пользователей авто-PC, а также других категорий пользователей. В предпочтительном варианте базовая структура поверхности предоставляет следующие услуги.

1. Методы открытия семантического запроса на основе SQLML. Это может быть локальным SQLML-документом, агентом и т.д.
2. Методы открытия URL агента непосредственно.
3. Методы просмотра семантической среды информационного агента.
4. Методы взаимодействия с системным буфером обмена с использованием настраиваемых форматов буфера обмена.
5. Методы сохранения текущей поверхности для данного запроса или для ИД данного семантического класса.

Поверхности. Как отмечено выше, поверхности являются шаблонами представления, которые используются для настройки пользовательского опыта на основе для каждого агента. В предпочтительном варианте поверхности являются XSLT-шаблонами и/или сценариями, которые находятся на централизованном сервере. Поверхности согласно изобретению предпочтительно генерируют XHTML-TIME код (например, для отображения презентатора, режима текст-речь, структурированной векторной графики (SVG) посредством встроенных модулей и т.д.) и получают доступ к различным системным сервисам. В предпочтительном варианте поверхности поддерживают следующие функции.

1. Отображение некоторых или всех полей, соответствующих XML-схеме объекта(ов), которые отображаются. Поверхность дополнительно обеспечивает пользователям способ уникального различения объектов в возвращаемом наборе или предоставляет пользователям обычные средства доступа, например, имя файла, URL или персональное имя (для людей).

2. Отображение пользовательского интерфейса, указывающего, понятен ли объект для агентства, содержащего его. Каждый объект предпочтительно включает в себя поле «понятно», которое отражает эту информацию.

3. Для типа семантического объекта SEMANTICOBJECTTYPE_ОБЪЕКТ поверхность дополнительно отображает метаданные исходного объекта или отображает метаданные XML-схемы для объектов специфического класса, которые представляют исходные объекты. Для поверхностей, которые отображают специфическую для класса XML-схему для запросов, относящихся к исходным объектам, поверхности должны быть «интеллектуальными», чтобы отображать специфическую для класса информацию на разных панелях. Предпочтительными путями выполнения этого является использование фреймов, диалоговых окон с выбором клавишей табуляции или других методов пользовательского интерфейса. Поскольку каждый семантический запрос указывает на исходные объекты, поверхность предпочтительно либо загружает запрос с фильтром SEMANTICOBJECTTYPE_ОБЪЕКТ (который просто возвращает исходные объекты), либо ИД типа требуемого объекта. В предпочтительном варианте для подготовки представления списка объектов с исходными объектами множества классов, поверхность должна сначала получить запрос объекта,

для каждого типа семантического объекта определить, сколько объектов существует в ресурсе для данного типа объекта. Это предпочтительно получают посредством вызова метода XML-Web-сервиса агентства GetNumObjectsOfClassInAgent с URL агента и именем ИД типа объекта (электронная почта, документ, событие и т.д.) в качестве аргумента. XML Web-сервис возвращает число объектов в агенте, удовлетворяющих фильтру ИД типа объекта.

В зависимости от того, сколько типов объектов имеется в запросе агента, поверхность отображает фреймы или другой пользовательский интерфейс, который соответствует количеству типов объектов. В предпочтительном варианте, если поверхность готова загружать метаданные, специфические для типа

объекта, она вызывает способ ExecuteSemanticQuery (выполнить семантический запрос) XML Web-сервиса с URL агента в качестве аргумента.

4. Когда пользователи указывают объект, отображается больше метаданных для объекта.

5. Если выбрана интеллектуальная лупа интеллектуального агента, то информационный объект, соответствующий настоящему изобретению, отображает контекстные метаданные, которые отображают объект в интеллектуальной лупе с объектом рядом с мышью. В одном варианте интеллектуальная лупа применяется к объектам, отображаемым в презентаторе. В альтернативном варианте интеллектуальная лупа может быть вызвана в другие приложения (например, приложения Microsoft Office, на рабочий стол и т.д.). Это связано с установкой системных методов (ловушек, перехватчиков) для отслеживания действий мыши и активизации приложения интеллектуальной лупы, когда мышь перемещается куда-либо в системе. Такой «перехватчик» вызывается на все события мыши, то есть он будет перехватывать все события мыши. Интеллектуальная лупа может, альтернативно, вызываться асинхронно. В этом варианте каждый раз, когда презентатор отображает новые результаты, он проверяет буфер обмена, чтобы убедиться, имеется ли там какая-либо семантическая информация интеллектуальной лупы. В асинхронном варианте презентатор автоматически кэширует все результаты интеллектуальной лупы для всех объектов в его представлении. Он отображает пиктограмму рядом с каждым объектом, который он представляет, указывающую, что в ней имеется контекстно-зависимая связанная информация. В предпочтительном варианте пользователи могут вызвать интеллектуальную лупу для любого объекта в представлении.

6. Информация, вызывающая прерывание. Каждый объект предпочтительно отображает пользовательский интерфейс, указывающий, имеется ли «информация, вызывающая прерывание», относящаяся к данному объекту. Это семантический эквивалент «новостей, вызывающих прерывание». Пользовательский интерфейс предпочтительно представляется для указания критичности информации, но не должен быть чрезмерно интрузивным в случае пользователей, не желающих просматривать информацию. Например, пользовательский интерфейс может быть показан как пиктограмма, которая медленно мигает в углу окна отображения объекта. Когда пользователь указывает на пиктограмму, отображаются метаданные по «информации, вызывающей прерывание». В предпочтительном варианте «информация, вызывающая прерывание» реализуется неявным специальным агентом, который активизирует вызовы ко всем агентам, использующим шаблон контекста новостей, вызывающих прерывание.

7. Каждый объект предпочтительно отображается с пользовательским интерфейсом, указывающим, имеет ли объект какие-либо аннотации. Эта информация включается как поле во все результаты запроса для всех объектов.

8. Предпочтительно каждый объект отображается с пользовательским интерфейсом, указывающим, имеется ли связанная информация по какому-либо предварительно определенному шаблону контекста или специальному агенту у клиента. Это предпочтительно включает специальных агентов, созданных пользователями, а также установленных по умолчанию специальных агентов (например, установленных клиентом). В предпочтительном варианте палитры контекста для шаблонов контекста отображаются для того, чтобы пользователь имел возможность выбора в отображении одной или более палитр контекста, в их скрытии, прокручивании (для перемещения по палитрам контекста) и т.д. Шаблоны контекста и палитры контекста описаны более детально ниже. В альтернативном варианте приоритеты агентства предпочтительно включают следующее.

Критический приоритет - наивысший приоритет. Например, для данного документа этот флаг будет иметь значение ИСТИННО (на агентстве), если связанное сообщение электронной почты было только что послано (в данном примере в пределах нескольких минут) или если имеется наступающее событие, которое неизбежно.

Высокий приоритет - приоритет, следующий за наивысшим. Обратная связь с пользовательским интерфейсом ясно указывает, что приоритет является достаточно высоким, чтобы служить основанием для обращения внимания пользователя, хотя обратная связь не должна быть излишне интрузивной. Приоритет является различным для разных пользователей, например, если имеется событие, локальное для пользователей, то его приоритет может быть выше, чем у события, удаленного (особенно, если для удаленного пользователя нет способа участия в этом событии).

Средний приоритет - этот приоритет просто указывает, что имеется информация, которую пользователь должен просмотреть, когда у него будет время. Обратная связь пользовательского интерфейса должна явно показать это.

Низкий приоритет - этот приоритет может указывать на то, что имеется связанная информация, которая уместна, но не является последней.

Виртуальные элементы сопряжения четырех приоритетов предпочтительно устанавливаются по умолчанию у клиента. Эти элементы сопряжения автоматически агрегируют информацию от соответствующих приоритетных агентов на каждом агентстве в списке «Мои агентства». На каждом агентстве имеются установленные по умолчанию приоритетные агенты. В предпочтительном варианте реляционные семантические запросы принимают во внимание контекст и пользовательскую информацию.

В предпочтительном варианте для каждого шаблона контекста (или текущего выбранного шаблона контекста) презентатор перечисляет агентства, которые пользователь добавляет в свой список «Мои

предпочтительные агентства», или последние агентства и запрашивает соответствующие агентства с использованием динамически генерируемого SQML для отыскания того, имеются ли какие-либо объекты, которые относятся к текущему объекту, на основе шаблона контекста. Если любое из агентств в списке предпочтительных или последних не доступно, то пользовательский интерфейс предпочтительно обрабатывает это прозрачным образом, игнорируя агентство. В предпочтительном варианте по умолчанию динамически генерируемые SQML-данные создаются путем индексации SQML-данных текущего выбранного SRML объекта и вставки ресурса в SQML в виде фильтра связи в SQML шаблона контекста (предпочтительно с использованием установленного по умолчанию предиката "relevant to" («релевантно»)). Это осуществляет интеллектуальную обработку отображения типа объекта текущего выбранного объекта на семантику отображаемой палитры контекста. Например, если текущий выбранный объект является документом, палитра контекста «Заголовки» использует SQML на основе вывода SQML для шаблона контекста «Заголовки». Каждое агентство в семантической области семантически обрабатывает результирующие SQML-данные, соответственно используя предикат по умолчанию. В другом примере, если выбранный объект является персоной, палитра «Заголовки» показывает заголовки, релевантные для персоны, например, «Заголовки», созданные или аннотированные этим лицом и т.д. Альтернативно, если текущий выбранный объект является документом или сообщением электронной почты, SQML (с предикатом по умолчанию) формирует семантические результаты, которые представляют семантически связанные заголовки по каждому агентству. Эти результаты предпочтительно отображаются в палитре контекста. То же самое применимо к другим палитрам контекста (например, классика, ньюсмейкеры и т.д.).

Для объекта «персона» флаг приоритета предпочтительно относится либо к объектам, к которым данная персона отослала сообщения, или к объектам, которые эта персона создала или располагает ими. В этом примере для осуществления такого определения предпочтительно используются только поля метаданных с семантической уникальностью (например, адрес электронной почты персоны).

9. Каждый объект предпочтительно отображает пользовательский интерфейс, включающий в себя ряд опций манипулирования. Например, пользовательский интерфейс, иллюстрирующий информационный объект, отображенный в панели результатов информационного агента (семантического браузера), показан на фиг. 54. Фиг. 54 показывает всплывающий контур (для метаданных объекта) и пиктограммы пользовательского интерфейса на объекте, позволяя пользователю активизировать опции инструментальных средств, таких как контекстная панель «Рекомендации», контекстная панель «Новости, вызывающие прерывание», всплывающее меню команд и т.д. Дополнительные и другие опции пользовательского интерфейса включают следующее:

внутренние семантические связи - это связи, которые являются присущими семантическому классу объекта. Если внутренние семантические связи отсутствуют, то ничего не требуется отображать. Например, объект электронной почты согласно предпочтительному варианту осуществления изобретения включает в себя следующие внутренние семантические связи:

1. Из списка →
 1. Персона А
 2. В список →
 1. Персона В
 2. Персона С
 3. Список точной копии →
 1. Персона D
 2. Персона E
 4. Список скрытой копии →
 1. Персона F
 2. Персона G
 5. Приложения →
 1. Документ 1
 2. Документ 2
 3. Документ 3

В предпочтительном варианте, если какие-либо из этих семантических связей активизируются пользователями, клиент осуществляет выборку метаданных для ассоциированного объекта (но не собственно объекта). Это позволяет пользователям анализировать семантическую информацию для разных аспектов исходного объекта. Поверхность предпочтительно вызывает XML-Web-сервис агентства, который содержит объект, с соответствующим методом. В предпочтительном варианте осуществления форма этого метода соответствует `SemanticRuntimeService::LoadNativeSemanticLink`. Этот вариант включает в себя ИД семантического класса, имя семантической связи, имя аргумента и форму строки аргумента. Например, для «перемещения» к третьему приложению (с индексом с отсчетом от нуля), поверхность должна вызывать `LoadNativeSemanticLink (SEMANTICCLASS_EMAILMESSAGE, "Attachments", "Index", 2)`. Это предпочтительно генерирует SQML, который представляет этот реляционный семантический запрос, создает нового временного интеллектуального агента, который имеет этот SQML, и загру-

жает интеллектуального агента. Это иллюстрирует предпочтительную семантическую навигацию. Процесс является факультативно рекурсивным. Пользователь может осуществлять перемещение от новых результатов с использованием любых новых объектов и опорных пунктов и т.д.

Пример всплывающего контура, ассоциированного с внутренними семантическими связями, соответствующий изобретению, показан на фиг. 55. В этом приведенном для примера пользовательском интерфейсе всплывающее меню отображается, когда пользователи выбирают пиктограмму «Внутренние связи» на информационном объекте в панели результатов. Эта иллюстрация показывает, какие внутренние семантические связи пользователи видят для объекта электронной почты. В предпочтительном варианте опции всплывающего меню активизируют новый SQML-запрос (относительно надлежащих связей ресурса и предиката), когда пользователи делают выбор соответствующей опции меню. Создается новый временный агент (с SQML), показывающий результаты запроса. Пользователи могут сохранить агента в своем списке предпочтений. Также новые результаты отражают внутренние семантические связи, шаблоны контекста и т.д., тем самым поддерживая управляемый пользователем просмотр ресурсов, при котором пользователи могут перемещаться семантически по информации. Альтернативная конфигурация и функциональные возможности для собственных команд следующие:

ALL INFORMATION (вся информация):

Найти связанную информацию на агентстве (только если она исходит от агентства)

Найти возможно связанную информацию на агентстве (только если она исходит от агентства)

Открыть аннотации →

Все

Аннотация 1

Аннотация 2

Аннотация 2

E-MAIL (электронная почта):

Из списка →

Персона А

В список →

Персона В

Персона С

Список точной копии →

Персона D

Персона E

Список скрытой копии →

Персона F

Персона G

Приложения →

Документ 1

Документ 2

Документ 3

PERSON (персона):

Сообщения к →

Прямые сообщения →

Член списка распределения →

Информация создана →

Информация аннотирована →

Информация с категориями, по которым данная персона является экспертом →

CUSTOMER (клиент):

Информация создана →

Аннотации. Это предпочтительно позволяет пользователям перемещаться для просмотра резюме по всем аннотациям для текущего объекта. В предпочтительном варианте поверхность отображает все аннотации путем вызова `ISemanticRuntimeService::EnumAnnotations` (с метаданными объекта в качестве аргумента). Это возвращает XML-представление таблицы свойств, содержащей метаданные для объектов аннотации. Поверхность предпочтительно отображает некоторое представление резюме текущей отображаемой аннотации (например, имена или названия аннотаций). Когда пользователями активизируется некоторая связь аннотации, поверхность отображает метаданные для объекта аннотации. Эти функции предпочтительно исходят от фильтров, применяемых на клиенте. Альтернативно, эти функции могут быть созданы как агент. Этот аспект настоящего изобретения дополнительно иллюстрирует семантическую навигацию. Аннотации предпочтительно загружаются с использованием SQML-представления запроса «Аннотации». Это создает нового интеллектуального агента с этим SQML. Интеллектуальный агент затем добавляется в список «последнего» и загружается (или просматривается). Процесс является факультативно рекурсивным. Пользователь может перемещаться с использованием новой(ых) отобра-

женной(ых) аннотации(й) в качестве опорного пункта и т.д.

Связанные объекты. В предпочтительном варианте это позволяет пользователям найти связанную информацию на каждом агентстве, включенном в пользовательский список «Мои агентства», с использованием текущего объекта в качестве опорного пункта информационного объекта. Это предпочтительно выполняется, не прибегая к копированию или вставке, или к пользовательскому интерфейсу расширения оболочки. В предпочтительном варианте всплывающее меню пользовательского интерфейса показывает информацию в следующем формате:

```
Find Related Objects →
All my agencies →
Agency Foo →
All. All
All.Understood.All
All.CriticalPriority.All
All.HighPriority.All
All.MediumPriority.All
All.LowPriority.All
All.MyFavorites.All
All.Recommended.All
Agencies that understand this object →
Agency Bar →
All.All
All.Understood.All
All.CriticalPriority.All
All.HighPriority.All
All.MediumPriority.All
All.LowPriority.All
All.MyFavorites.All
All.Recommended.All
```

Список "All my agencies" (Все мои агентства) получается презентатором просто путем перечисления агентств, которые пользователи зарегистрировали локальным образом. Презентатор возвращает список "Agencies that understand this object" (Агентства, которые понимают этот объект), «спрашивая» каждое локально зарегистрированное агентство, понимает ли оно конкретный объект. Презентатор пропускает XML-представление объекта в агентство, которое пытается семантически обработать XML-представление. Агентство возвращает флаг, указывающий, понимает ли он данный объект. Презентатор оптимизирует возвращенный список путем исключения агентства, на котором содержится сам объект, поскольку каждый объект имеет поле, которое указывает, понимает ли агентство его содержимое.

Команды действия (в форме глагольного императива). Эти команды позволяют пользователям активизировать любые действия, которые относятся непосредственно к текущему объекту. Например, документ или сообщение электронной почты может иметь команду действия "Open" (открыть). Эта команда открывает текстовый процессор или клиента электронной почты и отображает информацию. Событие может иметь команду "Add to Outlook Calendar" (добавить к календарю на перспективу). В предпочтительном варианте команды действия, предпочтительно специфические для классов, активизируются на клиенте базовой структуры системы. Агентству не требуется ничего знать о командах действий. В предпочтительном варианте имеются различные команды действия для каждого объекта. Эти команды действия предпочтительно отображаются сначала во всплывающем меню. В предпочтительном варианте команды действия включают следующее:

1. Annotate (аннотировать). Когда пользователь вызывает эту команду действия, поверхность предпочтительно осуществляет информационный обмен с рабочим циклом клиента и вызывает метод аннотирования. Этот метод инициирует установленного по умолчанию почтового клиента соответствующей предметной строкой (которую агентство подвергает синтаксическому анализу для интерпретации аннотации). Пользователи посылают регулярное сообщение электронной почты в качестве аннотации для объекта. Аннотации электронной почты факультативно включают приложения, которые также образуют семантические связи. Это позволяет пользователям перемещаться от объекта (например, документа) к его аннотации и затем к внешнему источнику содержимого (например, посредством интеллектуальной лупы). Альтернативные варианты осуществления также поддерживают аннотации, например, простые связанные с формой аннотации или диалоговые аннотации. Однако электронная почта обеспечивает наибольшее семантическое богатство.

2. Copy (копировать). Эта команда копирует XML-объект в системный буфер обмена.

3. Hide (скрыть). Это показывает, что пользователи не заинтересованы в просмотре объекта.

4. Open (открыть). Это определяется связью с тем, что открывается. В примере с документом может отображаться «Открыть документ». Для сообщения электронной почты может отображаться «Открыть

электронную почту». Клиент открывает объект посредством стандартного приложения, зарегистрированного в системе для MIME (многоцелевые расширения электронной почты) - типа связей. В альтернативном варианте настоящее изобретение поддерживает другие формы команды действия «открыть», такие как "Open with...", которые позволяют пользователям открыть объект конкретным приложением.

5. Mark as Favorite (маркировать как предпочтение). Это предпочтительно отображается, если агентство поддерживает пользовательское состояние и если объект не является предпочтением.

6. Unmark as Favorite (отменить маркировку как предпочтение). Это предпочтительно отображается, если агентство поддерживает пользовательское состояние и если объект является предпочтением.

Пример всплывающего контура, связанного с пользовательским интерфейсом команд действия, в соответствии с настоящим изобретением представлен на фиг. 56. В этом пользовательском интерфейсе всплывающее меню отображается, когда пользователи указывают пиктограмму "Verbs" (команды действия) на отображаемом информационном объекте в панели результатов. Меню показывает релевантные и поддерживаемые действия для информационного объекта на базе типа объекта (например, документ, электронная почта, персона и т.д.). Альтернативная конфигурация и функциональные возможности для собственных команд действия следующие:

Вся информация:

Аннотировать (Открывает Outlook; если объект из агентства, то адрес агента электронной почты агентства заполняется в поле "to" («к»), если нет, то поле "to" остается пустым, так что пользователь может указать агентство для ассоциирования аннотации объекта). Если объект не из агентства, то объект должен быть присоединен к сообщению электронной почты либо как URL, либо как полное приложение).

Копировать.

Открыть.

Маркировать как предпочтение (сохраняется у клиента).

Отменить маркировку как предпочтение.

Персона и клиент : «послать электронную почту».

Когда поверхность загружает новый запрос или метаданные для одного или более объектов, поверхность предпочтительно вызывает базовую структуру с запросом или метаданными. А в предпочтительном варианте поверхность не выполняет запроса, а направляет запросы в рабочий цикл презентатора, который затем управляет результатами.

11. Режим глубинной информации (или презентации).

Альтернативный вариант осуществления настоящего изобретения обеспечивает поддержку поверхности для режима глубинной презентации. В этом варианте поверхность отображает пользовательский интерфейс, указывающий, имеется ли связанная информация для текущего объекта. Поверхность также отображает текст, описывающий эту информацию. Например, для данного объекта документа поверхность может отображать всплывающее окно с текстом «Джейн Дое послала самое последнее сообщение электронной почты, которое относится к данному объекту: <резюме сообщения электронной почты>». В этом варианте поверхность показывает детали конкретной информации, такие как самый последний посланный связанный объект или самый ближайший наступающий объект. Поверхность может факультативно отображать другие «истинные» или логически выведенные данные, которые могут быть интересны для пользователей. Примеры включают следующее:

Лиза Хейлборн недавно послала связанный документ: <резюме>.

Наиболее вероятным автором данного документа является <foo> (некто).

Стив Джадкинс сообщает Патрику Шмитцу. Патрик послал 54 объекта критического приоритета, которые относятся к данному объекту.

Этот документ имеет 3 вероятных эксперта: <имена>.

Юинг Чен, по-видимому, имеет наибольшие экспертные знания по данному документу.

Базовая структура настоящего изобретения обладает различными уровнями «семантической глубины», которые поверхность использует для получения информации. Интеллектуальная лупа может также конфигурироваться для поддержки режима глубинной презентации. Иными словами, в предпочтительном варианте активизация интеллектуальной лупы на объекте возвращает глубинную информацию, подобную показанной выше. Поверхность показывает пиктограмму в углу окна отображения объекта. Пользователи могут щелкнуть по пиктограмме для отображения «глубинной информации». Метаданные для «глубинной информации» могут извлекаться асинхронным образом.

Пример всплывающего контура, связанного с пользовательским интерфейсом режима глубинной информации в соответствии с настоящим изобретением, показан на фиг. 57, как представлено в контекстной панели результатов. В этом примере пользователи имеют опцию выбора шаблона для глубинной информации, который фильтрует то, какой тип глубинной информации отображать, просмотра «предыстории» глубинной информации вместе с семантическими (SQML) связями с объектами, которые находятся в семантической среде (например, объект персоны «Стив Джадкинс», объекты результатов шаблона контекста «эксперты», объекты «прямые сообщения», использующие фильтр предиката «прямые сообщения») и т.д. Кроме того, пользователи имеют опцию предварительного просмотра результатов се-

мантических запросов «на месте» с использованием воспроизведения/управления предварительным просмотром.

е. Документ семантического запроса.

С точки зрения клиента, каждая вещь, которую он понимает, является документом запроса. В настоящем изобретении клиент открывает «документы запроса» путем, аналогичным тому, как текстовый процессор открывает «текстовые и составные документы». Клиент главным образом обеспечивает обработку документа семантического запроса и воспроизведение результатов. Документ семантического запроса предпочтительно выражается и сохраняется в форме языка разметки семантического запроса (SQML). Это подобно «семантическому формату файла». В предпочтительном варианте семантический формат файла SQML состоит из следующего:

Head (заголовок). Тег (признак) заголовка включает признаки, которые описывают документ.

Head:Title - это указывает на название документа.

Filters (фильтры). Презентатор фильтрует все возвращенные объекты с использованием записей в признаке "filters". Эти записи факультативно содержат имена типа объекта (документы, события, электронную почту и т.д.). Если фильтры не определены, то ни один объект не фильтруется. Тег имеет спецификатор, который указывает на то, должны ли записи быть включены или исключены. В случае избыточных записей (указанных обоими тегами "include" (включить) и "exclude" (исключить)) интерпретатор исключает записи (т.е. в случае препятствия предполагается признак «исключить»).

Attributes (атрибуты). Этот тег указывает атрибуты документа.

Skins (поверхности). Этот является вышестоящим (родительским) тегом для всех связанных с поверхностью записей.

skin:<objecttypename>. Это содержит информацию для поверхности для управления объектами типа объекта, указанного в «имени типа объекта». Презентатор использует стандартные поверхности и поверхности агента для объектов, которые не имеют соответствующих записей поверхности в SQML-документе. Опции предпочтительно включают следующее:

skin:<objecttypename>:color. Это содержит информацию о шаблоне цвета для использования в этом документе. Основной записью является XSLT URL.

skin:<objecttypename>:design. Это содержит информацию о шаблоне проектирования для использования в этом документе. Основной записью является XSLT URL.

skin:<objecttypename>:animation. Это содержит информацию о шаблоне анимации для использования в этом документе. Основной записью является XSLT URL.

Query (запрос). Это вышестоящий тег для всех основных записей запроса документа запроса и может включать в себя следующее:

Resource. Запрашивается ссылка на ресурс. Примеры включают пути файла, URL, идентификаторы записей кэша и т.д. Это будет отображаться интерпретатором на действительные компоненты администратора ресурсов.

resource:type. Данный тип ссылки на ресурс определяется пространством имен. Примерами определенных типов ссылки на ресурс являются nervana:url (это указывает, что ссылка на ресурс является хорошо определенным стандартным указателем ресурсов сети Интернет или специализированным URL, подобным "agent://...") и nervana:filepath (это указывает, что ссылка на ресурс является путем к файлу или каталогу в файловой системе).

resource:arg. Это указывает факультативную строку, которая будет направляться к ресурсу, когда интерпретатор преобразует ссылки на ресурс в действительные ресурсы. Это эквивалентно аргументу командной строки на исполняемом файле. Заметим, что некоторые ресурсы могут интерпретировать аргументы как часть gref (ссылки на ресурс), а не как часть gref argument (аргумента ссылки на ресурс). Например, стандартный URL может передать gref argument в конце самого URL (предпосялая вначале тегом «?»).

resource:version. (См. ниже).

resource:link. Все теги связи.

resource:link:predicate. Это указывает тип предиката для связи. Например, предикат nervana:relevantto указывает, что запрос соответствует: «возвратить все объекты из ресурса R, которые относятся к объекту O», где R и O являются определенными ресурсом и объектом соответственно. Другие примеры предикатов включают nervana:reportsto, nervana:teammateof, nervana:from, nervana:to, nervana:cc, nervana:bcc, nervana:attachedto, nervana:sentby, nervana:sentto, nervana:postedon, nirvana:containstext и т.д.

resource:link. Это указывает ссылку на объект семантической связи.

resource:link:type. Это указывает тип ссылки на объект, указанный в теге "oref". Примеры включают стандартные типы XML-данных, включая xml:string, xml:integer; специализированные типы, включая nervana:datetimeref (что может относиться к ссылкам на объект, подобным «сегодня» и «завтра»), и любые стандартные URL сети Интернет (HTTP, FTP и т.д.) или URL системы (objects://), которые относятся к объекту, который настоящее изобретение может обрабатывать как семантический XML-объект.

resource:link:version. Это указывает версию семантической связи ресурса. Это позволяет процессору семантической связи агентства возвращать результаты, которые имеют версии. Например, одна версия

семантического браузера может использовать V1 запроса, а другая версия может использовать V2. Это позволяет агентству обеспечивать совместимость в обратном направлении как на уровне ресурсов (например, для агентов), так и на уровне связи.

Query Type. Это указывает тип объектов, которые возвращает запрос (например, документы, электронная почта, заголовки, образцы и т.д.). Альтернативно, это может включать в себя имена типов объектов информации, шаблонов контекста и т.д.

Например, пример В в приложении иллюстрирует документ семантического запроса в соответствии с настоящим изобретением.

В предпочтительном варианте презентатор включает в себя SQML-интерпретатора. Когда презентатор открывает SQML-файл, он предпочтительно интерпретирует его сначала путем синтаксического анализа, проверки, создания таблицы записи мастера и затем исполнения записей в таблице записей. По существу, он «компилирует» SQML-файл перед его «исполнением», во многом подобно тому, как компилятор языка компилирует исходный код в объектный модуль, прежде чем он свяжет его затем с другими модулями и потом будет исполнять. В случае SQML-интерпретатора этот процесс факультативно связан с загрузкой других SQML-файлов посредством ссылок. Этот процесс предпочтительно не является циклическим. Клиент использует шаблоны XSLT в тегах "<skin>" (поверхность) (если доступны и не переопределяются по умолчанию поверхностями агента) для отображения информации для каждого объявленного типа объекта. Любые возвращаемые объекты, которые не имеют объявленной поверхности, отображаются с установленной по умолчанию поверхностью типа объекта или, в случае одиночной записи агента, с поверхностью агента (если она определена).

В альтернативном варианте клиент может загрузить новую поверхность для отображения каждого типа объекта даже после открытия документа семантического запроса. В этом варианте осуществления тег "<skin>" (поверхность) предпочтительно информирует клиента, с какой поверхностью загрузить запрос первоначально. В этом варианте определенная поверхность предпочтительно соответствует заявленному типу объекта.

В предпочтительном варианте осуществления базовая структура исполняет документ в двух фазах: фазе подтверждения действительности и в фазе исполнения. Для фазы подтверждения действительности интерпретатор сначала формирует таблицу семантических записей мастера. Таблица связывается с URL ресурса и имеет столбцы для оператора, ресурса, типа ресурса, предиката, типа предиката и связи. Интерпретатор исключает все избыточные записи по мере того, как он добавляет записи в таблицу. Также интерпретатор предпочтительно определяет все URL, прежде чем он добавит их в таблицу. Например, <http://www.abccorp.com> и www.abccorp.com/ интерпретируются как идентичные, поскольку они оба используют одну и ту же каноническую форму. Интерпретатор формирует и поддерживает отдельную таблицу SQML-ссылок. Эта таблица включает в себя канонический путь к SQML-файлу. Когда интерпретатор загружает исходный SQML-файл, он добавляет канонический путь к файлу в таблицу ссылок. Если SQML-файл указывает на самого себя, то интерпретатор игнорирует запись или возвращает ошибку. Если SQML-файл указывает на другой SQML-ресурс, то он добавляет новый файл в таблицу ссылок. Затем он рекурсивно загружает новый ресурс и процесс повторяется. Если в ходе обработки интерпретатор получает SQML-запись, которая уже содержится в таблице ссылок, то интерпретатор возвращает ошибку в вызывающее приложение (указывающую, что в SQML-документе имеется рекурсивный цикл). По мере того как интерпретатор находит дополнительные ресурсы в пути в графе документа, он добавляет их в таблицу записей мастера для данного ресурса. Он динамически добавляет связи для данного ресурса в запись этого ресурса в таблице записей. В результате интерпретатор эффективно выравнивает графы связей документа для каждого ресурса в графе.

Интерпретатор затем переходит к фазе исполнения. На этой фазе интерпретатор просматривает таблицу семантических записей и исполняет асинхронно, или последовательным способом, все запросы ресурсов. Затем он обрабатывает каждый ресурс на основе типа ресурса. Например, для файловых ресурсов, он открывает метаданные свойств для файла и отображает метаданные. Для HTTP ресурсов, которые относятся к «понятному» (согласованному) типу, например, к документам, интерпретатор загружает URL, выделяет их и отображает. Для ресурсов агента он запрашивает XML-Web-сервис по каждому агенту и передает связи как XML-аргументы, определяя каждую связь оператором. В предпочтительном варианте операторы для связей, которые пересекают границы документов, - это всегда оператор «И». Иными словами, интерпретатор будет всегда обрабатывать в соответствии с оператором «И» все связи для идентичных ресурсов, которые не объявлены вместе, поскольку рекурсивные запросы предполагаются отфильтрованными. Интерпретатор формирует столько вызовов компонента, представляющего ресурс, сколько имеется ресурсов агента. Для каждой связи интерпретатор разрешает связь путем преобразования ее в запрос, пригодный для обработки ресурсом. Например, агент со связью с атрибутами:

```
<predicate>nervana:relevantto</predicate>
<oref>c:\foo.doc</oref>
<oreftype>nervana:filepath</oreftype>
```

разрешается путем выделения XML-метаданных объекта (например, c:\foo.doc) и вызова XML-Web-сервиса для ресурса агента с XML в качестве аргумента. Это иллюстрирует, каким образом локальный

контекст разрешается в обобщенный (основанный на XML) запрос, который сервер может понять и который он обрабатывает.

Для оптимизации запроса XML-Web-сервис агентства представляет методы для передачи различных аргументов, определенных операторами («и», «или» и т.д.). Интерпретатор предпочтительно выдает одно обращение в XML-Web-сервиса для ресурса агента со всеми аргументами связи.

Сценарии реализации семантического запроса. Ниже приведены примеры сценариев, иллюстрирующие реализацию и обработку документов семантического запроса в соответствии с предпочтительным вариантом осуществления настоящего изобретения.

Сценарий 1. Загрузка SQML-документа. Клиент создает временный файл и записывает его в буфер, содержащий атрибуты простой, локальной HTML-страницы. Эта страница включает в себя компонент клиентской базовой структуры (например, управляющий элемент ActiveX, Java-апплет, режим Internet Explorer и т.д.). Эта страница инициализируется этим компонентом, открывающим SQML-файл и уникальный ИД, идентифицирующий экземпляр информационного агента. Компонент сам открывает SQML-файл. Иными словами, клиентская базовая структура сообщает встроенному модулю, какой SQML-документ запроса открыть. Встроенный модуль открывает документ семантического запроса путем интерпретации его, как описано выше.

Сценарий 2. Открыть документы. Клиент открывает стандартное диалоговое окно, которое позволяет пользователю выбрать файлы, которые должны быть открыты. Диалоговое окно инициализируется со стандартными расширениями файла документа (например, PDF, DOC, HTM и т.д.). Когда пользователи выбирают документы, диалоговое окно возвращает список всех открытых документов. Клиент создает новый SQML-файл и добавляет записи ресурсов с путями открытых документов. Новому SQML-файлу дается уникальное имя (предпочтительно на основе глобально уникального унификатора (GUID)). Поскольку это временный файл, имя предпочтительно не представляется пользователям. Затем метод переходит к сценарию 1, как описано выше.

Сценарий 3. Открыть папку в документах. Клиент создает SQML-файл (как описано выше) и инициализирует его с одной записью ресурса: file://<folderpath>?includesubfolders=(true|false). SQML-файл загружается (как в сценарии 1) путем перечисления всех документов в папке и отображения метаданных для документов.

Сценарий 4. Сохранить как агента. Клиент открывает диалоговое окно, позволяя пользователям установить имя агента. Клиент переименовывает агента в семантической среде (см. ниже), присваивая ему новое имя. Сохраняемый агент может быть временным или может быть уже сохраненным под другим именем. Информационный агент предпочтительно предлагает имя агента.

Сценарий 5. Сохранить в элементе сопряжения. Клиент открывает диалоговое окно, которое позволяет пользователям выбрать элемент сопряжения. Диалоговое окно предпочтительно позволяет пользователям создать новый элемент сопряжения. Когда элемент сопряжения выбран, клиент открывает SQML-файл элемента сопряжения в модель SQML-объекта и добавляет новую запись (текущий загруженный SQML-файл). Затем он дает приращение отсчету текущей записи.

Сценарий 6. Переместить и оставить. Клиент создает и открывает SQML-файл с единственной записью ресурса, например, подобной следующей:

```
<resource type="nervana:url">
agent://documents.all@abccorp.com
<linkpredicate="nervana:relevantto"
type="nervana:filepath"
c:\foo.doc
</link>
</resource>
```

В этом примере предполагается, что пиктограмму, представляющую "c:\foo.doc" перемещают и оставляют на пиктограмме в информационном агенте, ссылающемся на агента "agent://documents.all@abccorp.com."

Сценарий 7. Множественная операция «Переместить и оставить». Клиент создает и открывает SQML-файл с единственной записью ресурса, например, подобной следующей:

```
<resource type="nervana:url">
agent://documents.all@abccorp.com
<link predicate="nervana:relevantto"
type="nervana:filepath"
c:\fool.doc
</link>
<link type="nervana:filepath"
operator="or"
predicate="nervana:relevantto"
c:\foo2.doc
</link>
```

```

<link type="nervana:filepath">
operator="or"
predicate="nervana:relevantto"
type "nervana:filepath"
</link>
</resource>

```

В данном примере предполагается, что множество пиктограмм, представляющих "c:\foo1.doc", "c:\foo2.doc" и "c:\foo3.doc", перемещают и оставляют на пиктограмме в информационном агенте, ссылающемся на информационного агента "agent://documents.all@abccorp.com.". Также в этом примере предполагается, что пользователи указывают, что они хотят объединить семантические запросы, нацеленные на ресурсы агента.

Сценарий 8. Интеллектуальная лупа. Когда в информационном агенте выбрана интеллектуальная лупа, информационный агент указывает администратору семантической среды (см. ниже), что интеллектуальная лупа выбрана для идентификатора информационного агента. Когда поверхность уведомляет, что мышь находится над объектом (например, посредством события "onmouseover" в модели объекта документа (DOM)), она обращается сначала к презентатору, чтобы определить, находится ли информационный агент в режиме интеллектуальной лупы. Клиентская базовая структура определяет это путем запроса администратора семантической среды, находится ли информационный агент с идентификатором в режиме интеллектуальной лупы. Поскольку администратор семантической среды кэширует эту информацию из самого информационного агента, он может ответить на запрос от имени информационного агента. Если информационный агент находится в режиме интеллектуальной лупы, то клиентская базовая структура предпочтительно получает SQML-буфер от системного буфера обмена посредством администратора семантической среды. Это объясняется тем, что интеллектуальная лупа является виртуальной «вставкой», так как она получает свою информацию из буфера обмена. Иными словами, любой объект или агент, который копируется в буфер обмена, может быть использован как интеллектуальная лупа (даже регулярный текст). Базовая структура получает SQML-буфер и создает экземпляры компонентов ресурса для каждого ресурса в SQML-буфере. Клиентская базовая структура вызывает пользовательский интерфейс программирования приложения (API) ресурса GetInformationForSmartLens (получить информацию для интеллектуальной лупы), передающий XML-информацию для текущего отображаемого объекта на ресурс. Все ресурсы предпочтительно возвращают метаданные интеллектуальной лупы в клиентскую базовую структуру. Каждый ресурс предпочтительно возвращает метаданные в форме списка ядер информации интеллектуальной лупы. Каждое ядро содержит запись текста и список буферов запроса (в SQML). Запись текста содержит простой или специализированный текстовый формат, например, подобный следующему:

«Стив посылает к<A>Патрику. Патрик послал <A>54 сообщения критического приоритета, относящихся к данному сообщению».

Каждая пара тегов <A> предпочтительно включает в себя соответствующий буфер SQML-запроса в ядре информации. Клиентская базовая структура форматирует текст в DHTML (или иной подобный формат представления) для отображения в информационном агенте (например, в виде всплывающего контура или иного пользовательского интерфейса, предпочтительно чтобы не блокировать или отменять объект, над которым находится мышь). Клиентская базовая структура отображает пользовательский интерфейс для связей (аналогично HTML-связям), где найдены соответствующие теги "<A>" и "". Когда связь активизирована, клиентская базовая структура обращается к администратору семантической среды для создания новой записи кэша.

Администратор семантической среды указывает, в каком файловом пути должна быть сохранена запись. Клиентская базовая структура записывает SQML-буфер в файл соответственно тегу "<A>", на котором был выполнен щелчок. Клиентская базовая структура продвигает SQML-документ в администратор семантической среды и загружает SQML в информационного агента (посредством динамического HTML). Поскольку администратор семантической среды включает этот SQML-документ как текущий документ, пользователи могут сохранить документ посредством кнопки «сохранить» в информационном агенте (например, «сохранить как агента» или «сохранить в элементе сопряжения»). Пример информации, которую может отобразить интеллектуальная лупа, имеет следующий вид:

«Агент Email.Technology.All@Marketing имеет всего 300 объектов, которые относятся к данному объекту. Критический приоритет: 5 объектов, Высокий приоритет: 50 объектов, Средний приоритет: 100 объектов, Низкий приоритет: 145 объектов».

В предпочтительном варианте, если пользователи не выполняют щелчок на какой-либо из связей во всплывающем контуре, то никакой HTML-документ не создается, и ничего не добавляется к семантической среде. Это объясняется тем, что интеллектуальная лупа представляет только «потенциальный запрос». В предпочтительном варианте любая информация, которая может содержаться в SQML, может быть активизирована как интеллектуальная лупа (например, агенты, люди, документы, заголовки, образцы, агентства, текст, указатели URL HTTP-данных, указатели URL FTP-данных, файлы из файловой системы, папки из файловой системы, указатели URL электронной почты из приложений электронной поч-

ты, таких как Microsoft Outlook, указатели URL папок электронной почты и т.д.). Например, пользователи могут скопировать регулярный текст из текстовых приложений в буфер обмена. Если пользователи вводят информационного агента и выбирают интеллектуальную лупу, то SQML-версия текста будет выбрана как интеллектуальная лупа (посредством ресурса «документ»). Если затем над объектом «документ» выполняется указание «текстовая интеллектуальная лупа», то ресурс «документ», представляющий текстовую интеллектуальную лупу, дополнительно отображает показатель сходства, указывающий пользователям сходства между объектом «интеллектуальная лупа» и объектом, находящимся рядом с мышью. Если объект рядом с мышью является объектом «персона», то ресурс «документ» может принять решение запросить у агента, представляющего объект «персона», является ли агент экспертом по информации, содержащейся в тексте. Альтернативно, интеллектуальная лупа может отобразить связи со сходными документами или сообщениями электронной почты, созданными этой персоной, которые относятся к этому тексту.

Сценарий 9. Копировать и вставить.

Сору (копировать). При вызове команды «копировать» из семантической среды клиентская базовая структура копирует SQML-буфер в системный буфер обмена со специализированным форматом буфера обмена. Это обеспечивает то, что другие приложения (например, Microsoft Word, Excel, Notepad и т.д.) не распознают формат и не попытаются вставить информацию. SQML-буфер предпочтительно согласован с семантикой копируемого объекта. Например, операция копирования с объекта, отображаемого в презентаторе, осуществляется как копирование ресурса с соответствующим типом ресурса и URL ресурса, откуда получены метаданные. Копирование пиктограммы, представляющей агента, копирует URL агента или запись кэша, ссылающуюся на запись агента в семантической среде. Копирование информации из приложения рабочего стола (например, Microsoft Outlook) копирует SQML с типом ресурса, ссылающимся на исходное приложение и URL, указывающие на объекты в приложении. Эти URL предпочтительно отображаются в рабочем цикле интерпретатором на объекты в этом приложении. Например, копирование сообщения электронной почты из Outlook в семантическую среду может создать запись ресурса в следующем виде:

```
<resource type="nervana:outlookemailmessage">
  outlook://file://c:\temp\foo.html </resource>
```

Paste (вставить). При вызове команды «вставить» клиентская базовая структура создает SQML-файл на основе формата буфера обмена для вставляемой информации. Например, если буфер обмена содержит путь к файлу, то SQML-файл содержит связь (от ресурса, на котором была активизирована команда «вставить») к объекту с этим путем к файлу. Этот файл открывается, как описано выше. Если форматом буфера обмена является URL, то объектом является объект типа URL-объекта. Если формат соответствует регулярному тексту, то объект содержит действительный текст, в данном примере, тип ресурса nervana:text. Альтернативно, клиентская базовая структура создает запись кэша, сохраняет там текст (например, как .TXT файл) и сохраняет SQML-объект со ссылкой на путь к файлу и тип объекта, в данном примере nervana:filepath. Когда вызывается интерпретатор, он создает версию XML-метаданных текста и активизирует ресурс с аргументом XML-связи. Если формат буфера обмена является форматом SQML буфера обмена согласно изобретению, то выполняется сходный процесс, за исключением того, что если создается файл, то расширением будет .SQM (или .SQML). Это указывает интерпретатору, что объект является DQML-файлом, а не просто обычным текстовым файлом.

f. Семантическая среда.

Предпочтительный вариант осуществления настоящего изобретения обеспечивает представление каждого агента и агентства, доступного пользователю посредством информационного агента. Это предпочтительно включает в себя агентов, которые были сохранены локально в списке предпочтений «Мои агенты», недавно использованных агентов, агентов на локальных агентствах и агентов на удаленных агентствах. Удаленные агентства включают в себя агентства, которые оповещают о своем существовании через многоадресную передачу по локальной сети, агентства, доступные по глобальному каталогу агентств, и агентства, доступные через специализированные каталоги агентств. Агенты могут динамически добавляться к семантической среде посредством активизации их URL. В предпочтительном варианте иерархия семантической среды имеет структуру, показанную в примере С приложения. «Недавно использованные», «недавно созданные» агенты предпочтительно объединяются с «недавними (последними) агентами». Дополнительно могут быть добавлены «все агенты», «удаленные агенты» и «специализированный просмотр (представление)».

Представления (виды для просмотра) агентств позволяют пользователям видеть агентов в основном представлении посредством агентства. Представление по типу объекта позволяет пользователям просматривать тех же самых агентов, но отфильтрованных по типу объекта. Другие представления действуют аналогичным образом, например, «по контексту» (на основе шаблонов контекста) и «по времени». Семантическая среда объединяет запись «предпочтений» с записью «предыстории». Семантическая среда добавляет динамически управляемые представления, такие как «недавно использованные агенты» и т.д. Эти представления предпочтительно обновляются кодом, исполняемым в администраторе семантической среды (см. ниже).

Пример семантической среды, соответствующей настоящему изобретению, показан на фиг. 58 и 59. Пиктограммы, введенные в семантическую среду, могут включать в себя следующее:

- Приложение
- Все типы объектов контейнеров
- Все типы файлов документов
- Спецификатор пиктограммы агента новостей, вызывающих прерывание (например восклицательный знак)
- Спецификатор пиктограммы специального агента (например, ореол)
- Стандартный агент для каждого из типов объектов
- Агентство
- Контейнеры представлений агента
- Мои агенты
- Агенты новостей, вызывающих прерывание
- Предпочтительные агенты
- Специальные агенты
- Недавно использованные агенты
- Копии состояний объектов. Пользователи предпочтительно имеют возможность сохранять копии состояний (моментальные снимки) семантической среды. Копия состояния семантической среды, по существу, является временным кэшем состояния семантической среды. В предпочтительном варианте копия состояния включает в себя локально сохраненное состояние со следующей информацией:
 - Все агентства в момент моментального снимка, которые имеют новых агентов.
 - Время создания последнего агента для каждого агентства (на основе часов агентства).
 - Текущее время каждого агентства (на основе часов агентства).

Копии состояния предпочтительно доступны для пользователей. Информационный агент фильтрует семантическую среду, чтобы показать только агентства в списке копий состояния, и агентов в каждом из тех агентств, созданных между временем создания последнего агента и временем моментального снимка для каждого агентства.

g. Администратор семантической среды.

Настоящее изобретение обеспечивает администратора семантической среды, который предоставляет API для управления объектами семантической среды. В предпочтительном варианте управляемые объекты семантической среды содержат главным образом ссылки на агентов через SQML-буферы. Администратор семантической среды также предоставляет API для навигации в семантической среде. В предпочтительном варианте администратор семантической среды обеспечивает возможность экземплярам информационного агента:

1. Регистрироваться в администраторе семантической среды. Администратор семантической среды предпочтительно поддерживает информацию на всех открытых экземплярах информационного агента. Он делает это потому, что ряд сервисов (например, доступ к буферу обмена, доступ к интеллектуальной лупе и т.д.) выполняются в приложениях, таких как приложение расширения оболочки и компонент презентатора, исполняющихся в рамках управления браузером. Например, когда презентатор загружает новый SQML-документ в область отображения, ему необходимо получить запись кэша из администратора семантической среды. Он запрашивает администратора семантической среды о создании новой записи кэша для данного SQML-буфера. Администратор семантической среды создает запись кэша, записывает SQML-буфер в путь файла, соответствующий этой записи, создает временный HTML-файл, инициализируемый посредством управляющего элемента ActiveX, динамического режима HTML, Java-апплета (или эквивалентного клиентского механизма рабочего цикла), указывающего на запись кэша, и возвращает идентификатор записи кэша и файловый путь к временному HTML-файлу к презентатору. Например, в предпочтительном варианте временный HTML-файл может получить следующее имя:

c:\windows\temp\nervana_39fc54bc-81e5-4954-8cef-dla54935a0d.htm, где 39fc54bc-81e5-4954-8cef-3d1a54935a0d - идентификатор записи кэша. Содержащийся информационный агент автоматически обнаруживает новые загружаемые документы (посредством событий в управлении содержащимся информационным агентом). Содержащийся информационный агент может реагировать, когда пользователи выбирают опцию «сохранить» (например, «сохранить как агента» или «сохранить в элементе сопряжения»). Информационный агент выполняет это путем получения текущего пути к файлу документа, получения идентификатора записи кэша из пути к файлу (поскольку путь к файлу частично именован идентификатором), и отображения метаданных для записи кэша (имя, описание и т.д.), когда пользователи выбирают опцию «сохранить как». Информационный агент дополнительно запрашивает администратора семантической среды о повторном сохранении записи кэша с новым именем. Информационный агент регистрируется (предпочтительно при запуске) у администратора семантической среды с ИД процесса своего экземпляра. Администратор семантической среды выделяет новый идентификатор для информационного агента и сохраняет метаданные для экземпляра информационного агента (например, находится ли он в текущий момент в режиме интеллектуальной лупы). Информационный агент сохраняет этот идентификатор. Информационный агент предпочтительно передает идентификатор к администратору

семантической среды каждый раз, когда он осуществляет обращение. Информационный агент инициализирует презентатора с этим идентификатором. В предпочтительном варианте клиентская базовая структура вызывает администратора семантической среды с этим идентификатором каждый раз, когда ей требуются серверы в рамках нескольких приложений. Администратор семантической среды сохраняет идентификатор процесса экземпляра информационного агента, чтобы «собрать мусор» (утилизировать освобождаемую область памяти) по всем записям информационного агента, когда процессы информационного агента завершаются. Администратор семантической среды предпочтительно выполняет это, чтобы удалить запись информационного агента, поскольку информационный агент может «не знать», когда он завершается.

2. Добавить новые ссылки агента на семантическую среду. Записи ссылок агента предпочтительно сохранены в базе данных, файловой системе или системной памяти (например, регистре Windows). В предпочтительном варианте каждая запись семантической среды содержит:

- a. Идентификатор - уникальным образом идентифицирует агента в семантической среде.
- b. Имя - указывает имя агента. Информационный агент устанавливает по умолчанию имя агента, когда создается новый агент. Это имя агента устанавливается на основе способа создания. Например, если документ "foo" («нечто» метасинтаксическая переменная) копируется и вставляется поверх "bar" («что-то» - вторая метасинтаксическая переменная) агента, то информационный агент может создать временного агента, именуемого как "bar", относящийся к "foo" (в текущее время). Текущее время сохраняется для уникального именованного агента (в случае, когда пользователи повторно выдают тот же запрос). Пользователи имеют возможность переименовать агента, как желательно.
- c. Буфер запроса - указывает на буфер, содержащий SQL для агента.
- d. Тип - указывает тип агента (например, стандартный агент, элемент сопряжения, агент поиска, специальный агент и т.д.).
- e. Время создания - указывает, когда была создана запись агента.
- f. Время последней модификации - указывает, когда запись агента была последний раз модифицирована.
- g. Время последнего использования - указывает, когда запись агента была последний раз использована.
- h. Отсчет использования - указывает число раз использования агента, как автономного, как фильтра или как интеллектуальной лупы.
- i. Атрибуты - атрибуты агента (например, нормальный, временный, виртуальный, маркированный для удаления). Если запись временная, это означает, что пользователи не сохраняли ее в явном виде как локального агента. Временные записи предпочтительно используются в случаях, когда пользователи составляют комплексные запросы с использованием метода «перетащить и оставить», но без сохранения любого из промежуточных запросов как агентов. Когда пользователи сохраняют запрос как агента, информационный агент сбрасывает флаг временности, указывая на то, что запись запроса теперь является постоянной.
- j. Отсчет ссылок - указывает число ссылок на агента другими агентами и элементами сопряжения. Отсчет инициализируется на 0, когда создается новая запись агента.

3. Удалить агентов из семантической среды. Это предпочтительно выполняется в две фазы. Агенты могут маркироваться для удаления, и в этом случае администратор семантической среды устанавливает флаг, указывающий, что запись агента находится в «корзине». Запись агента может также быть удалена постоянным образом, и в этом случае запись удаляется из кэша окончательно.

4. Изменить свойства агента в семантической среде (например, сбросить временный флаг для агента, когда пользователи сохраняют агента).

5. Переименовать агентов в семантической среде.

6. Перенумеровать кэш для извлечения записей предпочтительно соответственно следующему:

- a. Все агенты
- b. Удаленные агенты
- c. Наиболее часто используемые агенты
- d. Используемые самыми последними агенты
- e. Созданные самыми последними агенты
- f. Фильтры для каждого типа объекта под вышеупомянутыми представлениями (например, документы, электронная почта, события и т.д.)
- g. Фильтры для агентств, которые содержат агентов в вышеупомянутых представлениях, фильтры для типов объектов на агентствах, и агентов, которые соответствуют этим представлениям (например, документы, электронная почта и т.д.)
- h. Фильтры для специальных агентов, основанных на шаблоне контекста (например, заголовки, образцы, ньюсмейкеры и т.д.). Примеры этих перечислений и представления показаны на фиг. 12-14, 17-19, иллюстрирующих представление дерева семантической среды.

7. Отфильтровать список агентов на основе счетчиков, обновленных посредством вызовов из экземпляров информационного агента. Каждый экземпляр информационного агента предпочтительно осу-

ществляет связь с одним администратором семантической среды. Таким образом, обновления носят ориентированный на пользователя, а не ориентированный на сеанс характер. Например, если пользователи открывают агента в одном информационном агенте, запись агента будет появляться в представлении последнего использованного агента в другом информационном агенте. Администратор семантической среды поддерживает информацию о числе раз использования каждого агента, последнем времени использования каждого агента и т.д. Он отфильтровывает агентов. Например, наиболее часто используемые агенты отфильтровываются на основе N агентов с наивысшим отсчетом использования, где N настраивается и где фильтр применяется только после некоторого периода стабилизации (например, после того как общий счет использования стал, по меньшей мере, Y, где Y также настраивается, например, на основе простой эвристики, такой как ожидаемое число использований агента на двухнедельный период). Недавно использованные агенты фильтруются на основе времени использования (которое сохраняется на основе по каждому агенту и которое обновляется экземплярами информационного агента каждый раз, когда агент используется). Недавно созданные агенты фильтруются на основе времени создания агента. Удаленные агенты фильтруются путем анализа флага маркировки для удаления по каждому агенту. Предпочтительные агенты фильтруются путем анализа флага маркировки как предпочтительных по каждому агенту. Для каждого из упомянутых исходных представлений соответствующие базовые представления заполняются с использованием простых фильтров. Представление агента заполняется путем анализа каждого возвращенного агента в исходном представлении и выделения из него уникальных агентств. Представления типа объекта под каждым из агентств отображаются в нем и затем заполняются путем фильтрации агентов на основе типа объекта агента (например, документ, электронная почта, событие и т.д.). Представление элемента сопряжения фильтруется путем отображения только агентов, которые имеют тип «элемент сопряжения». Представления типа объекта непосредственно фильтруются с использованием типа объекта агента. Представление «мои агентства» отображает локальные агентства. Каждое представление под ним предпочтительно является представлением типа объекта, отфильтрованным с использованием каждого доступного агента в агентстве. Представление «по контексту» заполняется путем фильтрации только специальных агентов (предпочтительно созданных с использованием шаблона контекста) и проверки имени контекста (например, заголовки, образцы и т.д.).

8. Поддерживать опорный отсчет агентов в семантической среде. Обязанностью вызывающего компонента (информационного агента) является приращение и уменьшение (отрицательное приращение) опорного отсчета записи документа. Информационный агент предпочтительно выполняет это посредством операций «перетащить и оставить», «копировать», «вставить» и т.д., иными словами, действий, которые создают новые запросы, относящиеся к существующим агентам.

9. Очистить семантическую среду - удаляет всех агентов.

10. Выполнить «сборку мусора». Администратор семантической среды автоматически удаляет всех устаревших (и временных) агентов. Кэш может быть конфигурирован для сохранения предыстории агентов до определенного возраста. Например, если кэш конфигурирован так, чтобы поддерживать информацию только в течение двух недель существования агентов, то он периодически выявляет временных агентов, возраст которых больше, чем две недели. Если он находит каких-либо таких агентов, то он автоматически удаляет записи агента, имеющие нулевой опорный отсчет. Это предпочтительно возникает в случаях, когда информационный агент создает новую запись кэша, но не создает другой записи (агента или элемента сопряжения), которая ссылается на нее. Иными словами, информационный агент выполняет отслеживание связи для непосредственной связи (чтобы избежать сложности). Администратор семантической среды дополнительно выполняет глубинную сборку мусора. Это происходит периодически на основе настраиваемого графика. Это применяется к записям, которые имеют опорный отсчет больше чем нуль, но не имеют действительных ссылок, поскольку связи не поддерживались, когда другие записи удалялись. Эта функция введена в предпочтительный вариант осуществления для минимизации сложности, поскольку информационный агент предпочтительно не отслеживает ссылки между агентами и элементами сопряжения, когда агенты и элементы сопряжения сохраняются или редактируются. В альтернативном варианте презентатор выполняет неактивное отслеживание связи агента, когда агент вызывается. Клиентская базовая структура игнорирует все ссылки, которые были удалены из семантической среды, аналогично тому, как Web-страница возвращает ошибку 404 (файл не найден), когда одна из ее связей была удалена. Иными словами, настоящее изобретение предусматривает ситуацию незавершенных запросов. Например, возможный сценарий может быть следующим:

элемент сопряжения B1 → ссылается на элемент сопряжения B2 → ссылается на агента A1 → ссылается на агента A2.

В этом случае опорный отсчет каждой записи может быть 1, даже если опорный отсчет цепочки равен 4. В общем случае, возможно иметь устаревшие записи, даже если опорный отсчет больше нуля. Для каждой записи, в отношении которой проводится операция сборки мусора, сборщик мусора отыскивает любую ссылку на запись во всех SQML документах. Если ни одной ссылки не найдено, то запись удаляется (если она временная или ее возраст превысил предельный возраст).

11. Манипулировать управлением уведомлениями. Пользователи могут регистрироваться для получения уведомлений от любого агента в семантической среде (например, сохраненных или локальных

агентов, стандартных агентов, элементов сопряжения т.д.). В предпочтительном варианте способы уведомления включают посылку электронной почты, сообщений реального времени, сообщений на пейджер, телефонных сообщений и т.д. Администратор семантической среды включает в себя администратора уведомлений (см. ниже), который будет управлять запросами уведомлений от пользователей через информационного агента. Администратор уведомлений сохраняет список требований на уведомления. Требование на уведомление предпочтительно включает в себя ИД объекта семантической среды (который идентифицирует агента), тип уведомления (электронная почта, сообщение реального времени и т.д.) и получателя, например, адрес электронной почты и т.д. Администратор уведомлений периодически опрашивает каждого агента в списке требований на уведомления, чтобы узнать, имеются ли новые объекты. Администратор уведомлений также передает «последнее запрошенное время» (на основе часов агента получателя). Агент отвечает числом новых объектов (путем активизации сохраненного запроса и передачи числа объектов в результаты запроса, которые были сохранены с «последнего запрошенного времени»). Агент отвечает текущим временем (по своим часам). Администратор уведомлений сохраняет время агента, чтобы избежать проблем с синхронизацией времени. Альтернативно, клиент и все агентства используют тот же самый сервер времени (Web-сервис времени) для получения своего времени, чтобы гарантировать то, чтобы все сравнения во времени осуществлялись на одной и той же шкале.

Каталоги агентств. В предпочтительном варианте администратор семантической среды предпочтительно поддерживает список агентств для каждого «каталога» агентства. Сеть многоадресной передачи предпочтительно выглядит для администратора семантической среды как каталог агентств. В предпочтительном варианте имеет установленный по умолчанию глобальный каталог агентств, сконфигурированный с использованием URL для XML-Web-сервиса на общедоступной системе. Этот XML-Web-сервис сохраняет кэш всех зарегистрированных агентств (предпочтительно с информацией, описанной выше, включая идентификатор, URL и т.д.). XML-Web-сервис предоставляет методы для обеспечения возможности агентствам зарегистрировать свое присутствие на каталоге агентства. XML-Web-сервис фильтрует избыточные записи. XML-Web-сервис также предоставляет методы для обеспечения возможности пользователям перенумеровать все агентства в каталоге агентств. Администратор семантической среды перенумеровывает каталог таким образом. Предпочтительно информационный агент рассматривает каталог агентства как расширение семантической среды и позволяет пользователям просматривать и открывать агентов в агентстве, перечисленном в каталоге агентств. Пользователи предпочтительно имеют возможность добавлять URL к специализированным каталогам агентств, которые могут быть установлены на внутренней сети.

Настоящее изобретение предусматривает создание и интеграцию настраиваемых каталогов агентств. Это по существу является альтернативой использованию многоадресной передачи для обнаружения в случаях, когда многоадресная передача не может быть реализована в сети (по причинам сохранения ширины полосы) или если некоторые подсети в сети широкого охвата не поддерживают многоадресную передачу.

h. Браузер среды (семантический браузер или информационный агент).

Браузер среды или информационный агент содержит обычный компонент Web-браузера (такой как управляющий элемент Internet Explorer ActiveX) и главным образом обеспечивает прием SQML-файла и визуализацию результатов через презентатора. В предпочтительном варианте он делает это путем открытия локального HTML-файла, инициализированного ссылкой на запись кэша документа SQML данного SQML-файла. HTML-файл загружает презентатор с использованием управляющего элемента (например, ActiveX, Java, режима Internet Explorer и т.д.). Этот элемент управления извлекает SQML-документ из кэша (посредством администратора семантической среды) и загружает SQML-файл, как описано выше. Управляющий элемент добавляет объекты к модели объекта документа (DOM) Web-браузера, когда он принимает обратные вызовы от ресурсов, указывающие, что объекты доступны для преобразования в HTML (или эквивалентный формат представления, предпочтительно через текущую XSLT и/или основанную на сценарии поверхность) и продвинуты в DOM для представления. Информационный агент позволяет пользователям открыть SQML-файл или запись в КЭШе (посредством ИД кэша). Информационный агент также позволяет пользователям перемещаться вперед и назад, и перемещаться по первому документу в стеке (аналогично опциями «вперед», «назад», «домой» в современных Web-браузерах, разница заключается лишь в том, что в этом случае SQML-документы открыты для интерпретации и отображения (результатов) в противоположность и другим документам).

На фиг. 60-68 представлены виды экранов информационного агента согласно предпочтительному варианту осуществления изобретения. На фиг. 60 показана семантическая среда, показывающая опцию всплывающего меню панели инструментов, позволяющие пользователю импортировать результаты локального поиска в семантическую среду, например, посредством неинтеллектуального агента, для создания нового специального агента, нового элемента сопряжения или нового локального агентства. Альтернативно, эти инструментальные средства могут быть собраны в одну кнопку инструментальных средств, которая активизирует мастера, из которого пользователи могут выбрать тип агента (неинтеллектуальный, интеллектуальный, специальный) или агентство, которое они хотят создать. На фиг. 61 показан примерный диалог, который позволяет пользователям осуществлять поиск в семантической среде с использова-

нием ключевых слов. Это создает нового интеллектуального агента (с соответствующим SQML). Пользователи предпочтительно имеют возможность настраивать имя нового интеллектуального агента и добавлять факультативное описание. На фиг. 62 показаны опции всплывающего меню инструментального средства «сохранить» панели инструментов, позволяющее пользователям сохранять вновь созданного или открытого агента постоянным образом в семантической среде (например, в списке «предпочтений») или сохранять агента в элементе сопряжения. На фиг. 63 показаны опции меню инструментального средства «интеллектуальная лупа» панели инструментов, которое позволяет пользователям вызывать интеллектуальную лупу (на основе интеллектуального агента или объекта, который в текущий момент находится в буфере обмена). Оно сообщает презентатору, что пользователю желательно использовать содержимое буфера обмена в качестве интеллектуальной лупы. Презентатор предпочтительно автоматически активизирует функциональные возможности интеллектуальной лупы для любого объекта, на который пользователь указывает (например, мышью). Меню также показывает опцию «вставить как интеллектуальную лупу и прикрепить», которая поддерживает интеллектуальную лупу включенной (даже в процессе перемещений по агенту), пока пользователь в явном виде не отключит опцию интеллектуальной лупы. На фиг. 64 показан пример представления диалога «открыть агента», показывающий, каким образом пользователи могут открыть серверных агентов из семантической среды и изменить «представление» среды (например, большие пиктограммы, малые пиктограммы, список и т.д.). Фиг. 65 показывает стандартный диалог Windows «открыть», показывающий, каким образом пользователь может импортировать «регулярный» документ из файловой системы в семантическую среду информационной нервной системы. Создается неинтеллектуальный агент, который ссылается на документ(ы). Когда неинтеллектуальный агент активизируется, документ(ы) открывается в информационном агенте и все семантические инструментальные средства (например, интеллектуальное копирование и вставка, шаблоны контекста и т.д.) активизируются для документа(ов). Это иллюстрирует, как браузер может сделать обычный «тупой» документ в файловой системе семантически «интеллектуальным». На фиг. 66 показан специализированный диалог «открыть документы в папке», который позволяет пользователям осуществлять поиск документов в папке в локальной файловой системе и импортировать их в семантическую среду. Это делает документы «интеллектуальными» за счет «представления» их через семантические инструментальные средства информационной нервной системы (например, интеллектуальное копирование и вставка, шаблоны контекста и т.д.). Фиг. 67 показывает диалоговое окно «просмотр папки», которое отображается, когда пользователи выбирают опцию просмотра. Это позволяет пользователям выбрать вариант открытия папки (из локальной файловой системы). Фиг. 68 показывает страницу из мастера «добавить элемент сопряжения», который позволяет пользователям выбрать, желательно ли им создать стандартный элемент сопряжения или виртуальный элемент сопряжения.

і. Дополнительные свойства приложения.

Расширения меню приложения и другие свойства базовой структуры. Клиент системы предпочтительно устанавливает расширение для приложений, которые поддерживают программные расширения, но которые уже не поддерживают копирование данных в буфер обмена. К ним относятся такие приложения, как Microsoft Windows Media Player и Microsoft Outlook (для заголовков сообщений электронной почты). В предпочтительном варианте расширения меню считывает «копировать». Система копирует выбранный объект как XML-объект в системный буфер обмена Windows. Например, системный встраиваемый модуль для Microsoft Outlook электронной почты копирует выбранный объект электронной почты как XML-объект электронной почты. Для приложений, которые уже поддерживают буфер обмена, не требуется расширения.

Серверные предпочтительные объекты. Агентства, которые поддерживают пользовательское состояние, могут маркировать объекты как «предпочтительные». Когда объект маркирован как «предпочтительный» презентатор активизирует метод на XML-Web-сервисе агентства. XML-Web-сервис добавляет семантическую связь между объектом пользователя и конкретным рассматриваемым объектом. В предпочтительном варианте осуществления пользователи могут просматривать предпочтительные объекты посредством устанавливаемого по умолчанию агента All.MyFavorites.All. Этот агент возвращает все объекты, которые были маркированы как предпочтительные. Администратор агентства может создать суб-агентов, например, All.MyFavorites.Technology.XML.All.

Презентатор позволяет пользователям маркировать и отменять маркировку предпочтений, что также является средством переопределения структуры, которую серверы и агентства экспортируют. Использование сценария «предпочтений» особенно ценно в случаях, когда пользователи могут просматривать объекты, представляющие интерес и не хотят обязательно перемещаться по ним. Свойство «предпочтений» может факультативно использоваться агентством, чтобы рекомендовать объекты пользователям. В предпочтительном варианте эти рекомендуемые объекты извлекаются посредством агента All.MyFavorites.All. Агентство рекомендует объекты, основываясь главным образом на объектах, которые пользователи маркировали как предпочтительные. Серверные предпочтения также будут предпочтительно использоваться с шаблонами контекста «предпочтения, образцы и рекомендации».

Хранители экрана агента. Предпочтительный вариант осуществления настоящего изобретения позволяет пользователям выбрать любого абонированного агента в качестве хранителя экрана. Пользовате-

лей предпочтительно предупреждают, что агенты могут характеризоваться чувствительными (зависимыми) данными, и предоставляют возможность определить, является ли надежным использование конкретного агента в качестве хранителя экрана. В предпочтительном варианте клиент системы может загрузить любого абонированного агента в качестве хранителя экрана. В альтернативном варианте хранителем экрана может быть структурированная поверхность, которая включает в себя параллельно отображаемых агентов, например, в четырех квадрантах экрана.

Интеллектуальная лупа из агента к агенту. В альтернативном варианте клиент системы поддерживает использование интеллектуальной лупы (активизируемой через агента или через элемент сопряжения) в качестве контекста для активизации другого агента или элемента сопряжения. Например, пользователи могут выбрать All.CriticalPriority.All и использовать этого агента в качестве интеллектуальной лупы для просмотра All.Understood.All, чтобы отыскать все объекты, которые имеют критический приоритет и которые также понятны агентству получателя.

Иллюстрации примеров пользовательских интерфейсов интеллектуальной лупы. На фиг. 69-71 приведены примеры всплывающего меню, связанных со свойством интеллектуальной лупы информационного агента согласно настоящему изобретению. На фиг. 69 показан пример всплывающего меню в контексте панели результатов интеллектуального агента в качестве интеллектуальной лупы. Этот пример показывает всплывающее окно, отображаемое, когда пользователи выбирают пиктограмму интеллектуальной лупы на информационном объекте. Этот пример показывает случай, когда интеллектуальный агент, озаглавленный как «документы Reuters, относящиеся к [My Nervana UI Specification] (моя спецификация пользовательского интерфейса Нервана), в буфере обмена и «посланы» в качестве интеллектуальной лупы на объекте электронной почты, озаглавленном "Yuying's Thoughts on the Nervana UI" (мысли Юйинга по поводу пользовательского интерфейса Нервана)». На фиг. 70 показан пример всплывающего меню в контексте панели результатов с объектом в качестве интеллектуальной лупы (и с «указанием» над агентом). Этот пример показывает, что интеллектуальная лупа имеет ассоциативную связь ($A[\text{SMART LENS}]B = B[\text{SMART LENS}]A$). Секция результатов панели контекста идентична показанной в примере на фиг. 69, указывающей, что интеллектуальная лупа имеет ассоциативную связь. На фиг. 71 показан пример всплывающего меню в контексте панели результатов с информационным объектом в качестве интеллектуальной лупы и информационным объектом в качестве предмета, который «рассматривается в лупу». В этом примере объект, озаглавленный как "My Nervana UI Specification" скопирован в буфер обмена (его SQL-представление) и вставлен как интеллектуальная лупа над другим объектом (в панели результатов), озаглавленным как "Yuying's Thoughts on the Nervana UI" (объект электронной почты). В этом примере пользователь имеет опцию выбора предиката, который семантически согласован с комбинацией документа и сообщением электронной почты. На фиг. 72 показан пример варианта всплывающего меню с фиг. 71, показывающий относительную меру двух объектов (объекта в качестве интеллектуальной лупы и объекта «рассматриваемого в лупу»), как в процентном соотношении, так и графически, в данном случае в виде столбчатой диаграммы. Фиг. 73-75 показывают примеры таблиц, которые иллюстрируют предикаты типов объектов «линии поведения» и «реляционное содержимое» с использованием интеллектуальных луп. На фиг. 73 показан сценарий агент-объект для всей информации, причем «линия поведения» интеллектуальной лупы является коммутативной, например, $A[\text{Интеллектуальная Лупа}]B = B[\text{Интеллектуальная Лупа}]A$. Фиг. 74-75 показывают сценарий объект-объект для документов и электронной почты, и здесь «линия поведения» интеллектуальной лупы является коммутативной, например, $A[\text{Интеллектуальная Лупа}]B = B[\text{Интеллектуальная Лупа}]A$.

Иллюстрации пользовательского интерфейса поверхности элемента сопряжения. На фиг. 76 показан пример пользовательского интерфейса, представляющий семантические результаты управляющего элемента плеер/предварительный просмотр.

Презентатор информационного агента предпочтительно присоединяет этот элемент управления к каждой панели результатов. Управляющий элемент плеер/предварительный просмотр обеспечивает возможность пользователям просмотреть результаты в панели результатов, анимировать результаты (воспроизведение, остановка, пауза, смена, скорость и т.д.) и отфильтровать результаты (например, в случае результатов элемента сопряжения). На фиг. 77 показан пример пользовательского интерфейса, иллюстрирующий семантические результаты элемента сопряжения. В этом примере поверхность элемента сопряжения имеет зарезервированные части области отображения как отдельные кадры для каждого агента в элементе сопряжения, и присоединенный управляющий элемент плеер/предварительный просмотр для каждого кадра, что позволяет пользователю индивидуально перемещаться, управлять и анимировать результаты по каждому агенту в элементе сопряжения. Альтернативно, поверхность элемента сопряжения может отображать объединенные результаты от всех агентов в элементе сопряжения (с одним присоединенным управляющим элементом плеер/предварительный просмотр), может отображать результаты в кадрах соответственно типу информационного объекта и т.д.

Множественная операция «перетащить и оставить». В альтернативном варианте клиент системы позволяет пользователям выбрать множество документов или папок с рабочего стола и использовать их как основу реляционных запросов на агента или элементе сопряжения. Это позволяет пользователям дополнительно уточнять запрос с использованием множества документов в качестве инструментального

средства уточнения. Например, пользователь может дополнительно указать, что ему желательно осуществит объединение или пересечение результатов (с использованием каждого из документов в качестве фильтра). Это создает SQML-файл с одним ресурсом (объектом, на который были «переташены» связи), и множеством связей (по одной на документ или на «переташенный» объект). Клиентский SQP предпочтительно интерпретирует это путем извлечения XML-метаданных для всех фильтров объекта и вызова XML-Web-сервиса интеллектуального агента получателя с XML-аргументами. В предпочтительном варианте XML-Web-сервис агентства осуществляет категоризацию аргументов XML-метаданных, формирует надлежащее SQL-представление запроса и возвращает результаты.

Соглашения о сокращенных URL. Агентства согласно настоящему изобретению могут совместно использовать Web-сеть Интернет, поскольку они факультативно инсталлируются как Web-приложения. В результате на агентства можно ссылаться с использованием схемы именования Web-сети (например, обычных HTTP URL). В предпочтительном варианте настоящее изобретение предлагает соглашения о сокращенных наименованиях и URL, которые являются специфическими для информационных агентов семантической среды.

Соглашение о сокращенном URL агента. Соглашение о сокращенном URL агента состоит в следующем:

```
agent://<agentname>@<agencyurl>?start=<start>&end=<end>&skin=
<skin url>
```

При активизации это предпочтительно отображается на полностью определенный HTTP URL, например,

```
http://<path to Agency ASP; или
CGIscript>?agentname=<agentname>& start=<start>&end=<end>&skin
=<skinUrl>.
```

Пример соглашения о сокращенном URL агента состоит в следующем:

```
agent://email.technology.wireless.all@marketing.abccorp.com?start=0&end=25&skin=http://www.nervana.
net/skins/email/abccema ilskin.
```

xslt

Этот URL разрешается клиентом следующим образом:

Запустить посредника Web-сервиса, открыть WSDL-файл

`http://abc.com/nervanaroot/webservice.wsdl` и запросить Web-сервис о статистике агентства с именем "Маркетинг". Для HTTP-доступа это будет отображаться на путь к ASP или CGI. Например:

```
http://abccorp.com/marketingagency.asp?urltype=agent&agent name=
email.technology.wireless.all&start=0&end=25&skin=http://www.nervana.net/skins/email/abccorp
emailskin.xslt
```

Начальный аргумент указывает начальный индекс с отсчетом от нуля для объекта, возвращаемого первым. Конечный аргумент указывает конечный индекс. URL поверхности является факультативным. Если никакой URL поверхности не определен, то клиент загружает агента с установленной по умолчанию поверхностью агента.

Доступ к локально сохраненному агенту может быть получен следующим образом: `agent://<agentname>@localhost`. Например, `agent://Documents.[Related to My Business Plan] @localhost` будет загружать локально сохраненного агента (в «моих агентах») с именем "Documents.[Related to My Business Plan]" («Документы.[относятся к моему бизнес-плану] »).

Соглашение об URL агентства. Примером может служить следующее:

```
agency://<agencyname>.<domainname>?query=getproperties|get stats|
getagents@agentviewfilter=<agentviewfilter>&
agentnamecontainsfilter=<agentnamecontainsfilter>&agenttype filter=
<agenttypefilter>&agentobjecttypefilter=<agentobjecttypefilter>
```

В этом примере аргумент запроса есть "getproperties". URL извлекает свойства самого агентства (например, имя, имя отображения, является ли оно локальным или удаленным и т.д.). Альтернативно, если свойство есть "getstats", то URL извлекает статистику агентства (полное число агентов, число стандартных агентов, число составных агентов, число агентов предметных областей, полное число объектов, число объектов документов, число объектов электронной почты и т.д.). В предпочтительном варианте флаг "getproperties" является установленным по умолчанию, означая, что эти свойства извлекаются, если никакой другой аргумент не определен. Если определены аргументы "getproperties" или "getstats", предпочтительно совместно не определяются никакие иные аргументы.

Аргумент "agentviewfilter" является факультативным и позволяет вызывающему определить представление агента для ограничения поиска. Например, представление агента "Reuters News" может быть инсталлировано на сервере для того чтобы только возвращать агентов, которые управляют объектами «новости» от Reuters. Аргумент "agentnamecontainsfilter" является факультативным и позволяет пользователю фильтровать результаты посредством строки поиска для имени агента. Аргумент "agenttypefilter" является факультативным и позволяет пользователю фильтровать агентов на основе типа агента (стандартный агент, составной агент или агент предметной области). Аргумент "agentobjecttypefilter" является

факультативным и позволяет пользователям фильтровать результаты типом объекта, которым управляет агент (электронная почта, документы, люди и т.д.). Примеры включают в себя следующее:

```
agency://sales.boeing.com?query=getstats (соответствует HTTP URL
http://boeing.com/salesagency.asp?urltype=agency&query=getstats)
agency://sales.boeing.com?agenttypefilter=standard&agentobj
jectypeidfilter=events (соответствует HTTP URL
http://boeing.com/salesagency.asp?urltype=agency&agenttype filter=
standard&agentobjectypeidfilter=events
```

Соглашение об URL объектов. Объекты агентств могут быть доступны непосредственно из клиента.

Соглашение об URL следующее:

```
objects://<querystring><agencyname><domainname>?querytype=
<objectid|searchstring>&objecttypefilter=<objecttypefilter>
```

Аргумент "objecttypefilter" является факультативным и позволяет пользователям фильтровать возвращаемые объекты типом объекта (электронная почта, документ, событие и т.д.). Примеры включают в себя следующее:

```
objects://34547848@support.attwireless.com?querytype=objectid возвратит объект с идентификатором
объекта (objectid) 34547848.
```

```
objects://80211 @support.attwireless.com?querytype=searchstring
&objecttype=email возвратит объекты электронной почты, согласованные со строкой запроса
"80211".
```

Соглашение об URL категории. Соглашение об URL следующее:

```
category://<<categoryname>@<kbsurl>?semanticdomainname=
<semanticdomainname>
```

Аргумент "semanticdomainname" является факультативным. В предпочтительно варианте, если он опущен, то выбирается устанавливаемая по умолчанию предметная область KIS. Примером является следующее:

```
category://technology.wireless.all@abccorp.com/marketing
knowledge.asp
```

Это соответствует категории "Technology.Wireless.All" (Технология.Беспроводная.Все) для устанавливаемой по умолчанию предметной области в базе знаний, инсталлированной на Web-сервисе abccorp.com/marketingknowledge.asp. Это будет отображаться на следующий HTTP URL:

```
http://abccorp.com/marketingknowledge.asp?category="technology.
```

wireless.all. Пример полностью определенной версии категории URL может быть следующим:

```
category://technology.wireless.all@abccorp.com/marketingkn
owledge. asp?semanticdomainname=
"/InformationTechnology"
```

Совместное использование и распространение клиентской информации. В предпочтительном варианте осуществления пользователи могут совместно использовать агентов (включая элементы сопряжения) с другими путем посылки им по электронной почте, в режиме передачи сообщений в реальном времени и т.д. Локальные пользователи информации предпочтительно способны либо сохранять информацию агента локально, либо использования информации при перемещении пользователя (например, посредством поддержки AbccorpliMirror в Windows 2000 режима роуминга в масштабах предприятия, посредством фирменного XML-Web-сервиса в глобальном каталоге агентства (с использованием паролей для идентификации), или посредством интеграции с Microsoft .NET My Services с использованием услуги идентификации Microsoft Passport).

Локальные агентства. Клиент системы предпочтительно также позволяет пользователям создавать и дополнять к списку «мои агентства» локальные агентства, которые исполняют локальный экземпляр KIS. В данном варианте клиент также позволяет пользователям удалять персональное агентство.

Согласованность и нераздробленность (целостность) пользовательского опыта. Информационный агент (семантический браузер) настоящего изобретения обеспечивает согласованный и целостный пользовательский опыт. Иными словами, информационный агент сосуществует непрерывным образом с современным браузером.

Инструментальные средства, такие как «назад», «вперед», «домой», «стоп», «обновить», «печать» предпочтительно работают так же, как в современном браузере, чтобы не вводить в заблуждение пользователя. Многие из инструментальных средств остаются теми же, хотя их функциональные возможности отличаются. Кроме того, новые инструментальные средства предпочтительно добавлены к панели инструментов и опциям меню, отражая новые функциональные возможности в семантическом браузере (их можно видеть на панели инструментов на экранных изображениях). Фиг. 78 и 79 иллюстрируют приведенные для примера отображения функциональных возможностей согласно настоящему изобретению, иллюстрирующие предпочтительные отображения для введения новых функциональных возможностей для пользователей при сохранении согласованности модельных представлений. На фиг. 78 показано сравнение устанавливаемых по умолчанию наборов инструментальных средств пользовательского интерфейса для современного Web-браузера и предпочтительного варианта информационного агента на-

стоящего изобретения. На фиг. 79 показано сравнение устанавливаемых по умолчанию наборов инструментальных средств пользовательского интерфейса для средства просмотра документов, используемого в Microsoft Explorer, и предпочтительного варианта информационного агента настоящего изобретения.

5. Обеспечение контекста в настоящем изобретении.

а. Шаблоны контекста.

Настоящее изобретение предусматривает шаблоны контекста, то есть управляемые сценарием шаблоны информационных запросов, которые отображаются на конкретные семантические модели для доступа к информации и ее извлечения. По существу, шаблоны контекста можно представить как «каналы» извлечения персональной цифровой семантической информации, которые доставляют информацию к пользователю путем использования предварительно определенного семантического шаблона. В предпочтительном варианте семантический браузер позволяет пользователю создавать нового «специального Агента» с использованием шаблонов контекста для инициализации свойств агента. Шаблоны контекста предпочтительно агрегируют информацию на одном или более агентствах.

Например, настоящее изобретение определяет описанные ниже шаблоны контекста. Дополнительные шаблоны контекста, направленные на интеграцию и распространение различных типов семантической информации также входят в объем настоящего изобретения (примеры включают шаблоны контекста, относящиеся к эмоциям, например, «злой», «печальный» и т.д., шаблоны контекста для определения местоположения, мобильности, внешних условий, пользовательских задач и т.д.).

Шаблон контекста «Заголовки». Шаблон контекста «Заголовки» (и результирующий его специальный агент) может быть сопоставимым с персональной цифровой версией программы CNN "Headline News" (краткие новости) в том, каким образом он добавляет семантическую информацию. Шаблон контекста позволяет пользователю получить доступ к информации заголовков от одного или более агентств, отсортированной в соответствии с временем создания или публикации информации и настраиваемым интервалом времени, который определяет «свежесть» информации. Например, программа CNN "Headline News" отображает заголовки каждые 30 мин (круглосуточно). В предпочтительном варианте информационный агент 30 настоящего изобретения позволяет пользователям создать специального агента «Заголовки» с использованием следующих фильтров и параметров:

Опорные пункты информационного объекта. Результирующий элемент сопряжения показывает результат, который относится к этому объекту. Это оптимальный параметр. Если он не определен, то заголовки отображаются для всего агентства (без какой-либо фильтрации на основе объекта).

Предварительно определенный период «обновления», например, 30 мин, 1 ч и т.д.

Предикат. Он определяет, каким образом опорный пункт информационного объекта связывается с информацией, подлежащей извлечению. Примерами являются: «относится к», «возможно, относится к» (использует поиск на основе текста), «создано (автором является)» (в случае объекта «персона»), «возможно, автором является», «имеет опыт в» и т.д. Стандартный предикат «относится к» предпочтительно используется по умолчанию. Этот стандартный предикат разрешается агентством путем интеллектуального отображения его на конкретные предикаты.

Агентство(а). Сюда относятся агентства для проверки заголовков. По меньшей мере одно агентство должно быть определено, и здесь нет предела для числа агентств, которые могут быть определены. Пользователь может указать, следует ли использовать все агентства в списках «последних» и/или «предпочтительных» агентств.

Список категорий, например, "Technology.Wireless.All". Это действует как дополнительный фильтр для запроса.

В дополнение к «свежести» информации шаблон контекста «Заголовки» предпочтительно определяет, насколько «горячими» являются элементы результатов, чтобы осуществить ранжирование результатов. Это может быть выполнено путем запроса агентства о выявлении числа семантически связанных объектов в агентстве, что является хорошим индикатором того, является ли предмет конкретного запроса «горячей» новостью. Кроме того, возвращаемые объекты (или элементы информации) предпочтительно сортируются по «свежести» или по их новизне.

Например, пример D в приложении иллюстрирует SQL-выходной результат с шаблона контекста «Заголовки» в предпочтительном варианте осуществления. В этом примере шаблон контекста извлекает всю информацию из четырех различных агентств (маркетинг, исследования, продажи, человеческие ресурсы). В предпочтительном варианте, в данном примере SQL, как для всех шаблонов контекста, можно дополнительно сформировать основу интеллектуальной лупы, интеллектуальных операций «копировать и вставить», «перетащить и оставить» и других инструментальных средств в семантической инструментальной панели.

Шаблон контекста «Новости, вызывающие прерывание». Шаблон контекста «Новости, вызывающие прерывание» (и его получаемый в результате агент) может быть сопоставлен с персональной цифровой версией вставок программы новостей CNN, которые прерывают регулярным образом спланированные программы, в том, как он передает семантическую информацию. Подобно вставкам новостей CNN, этот шаблон контекста позволяет пользователям получать доступ к «вызывающей прерывание», критичной к времени информации от одного или более агентств, предпочтительно отсортированной по времени

создания или публикации информации или по времени возникновения (в случае события), и с настраиваемым интервалом времени, который определяет «свежесть» информации, и с настраиваемым «предельным сроком» для событий, чтобы определить критичность к времени. Например, шаблон контекста может быть определен для фильтрации информационных объектов, посланных за последний час, или событий, которые должны состояться на следующий день.

В предпочтительном варианте осуществления шаблон контекста «Новости, вызывающие прерывание» отличается от агентов новостей, вызывающих прерывание. Шаблон контекста является шаблоном, который определяет параметры статического запроса, который поступает к одному или нескольким агентам. Агент новостей, вызывающих прерывание, является любым интеллектуальным агентом, который пользователи могут создать, и является по существу создаваемым и настраиваемым пользователем. Например, специальный агент новостей, вызывающих прерывание, основанный на шаблоне контекста «Новости, вызывающие прерывание», может информировать пользователей об информационных объектах, посланных на последний час, или событий, которые должны состояться на следующий день, которые связаны с локальным документом (или любым другим локальным контекстом, если определено). Но агент новостей, вызывающих прерывание, в приеме оповещений «Событиях в беспроводной технологии, о которых оповещено сотрудником моего подразделения, которые должны состояться в Сиэтле или Портленде в течение последующих 24 ч, и которые относятся к данному документу на моем жестком диске». Агент новостей, вызывающих прерывание, обеспечивает для пользователей большую гибкость и персонализацию, чем шаблон контекста «Новости, вызывающие прерывание». Преимущество шаблона контекста «Новости, вызывающие прерывание», состоит в том, что он предпочтительно формирует основу внутренне обусловленных оповещений с использованием параметров, которые определяются как «обуславливающие прерывание» для типовых пользователей.

Шаблон контекста «Обсуждения». Шаблон контекста «Обсуждения» (и получаемый в результате специальный агент) может быть сопоставлен с персональной цифровой версией вставок программы «Дебаты» CNN, в том, как он передает семантическую информацию. Подобно программе CNN «Дебаты», которая использует обсуждения и дебаты в качестве контекста для распространения информации, в предпочтительном варианте осуществления специальный агент «Обсуждения» отслеживает почтовые отправления, аннотации и потоки для релевантной информации. Шаблон контекста «Обсуждения» можно представить себе как шаблон контекста «Заголовки», отфильтрованный типом объекта «электронная почта». В дополнение к параметрам «Заголовков», шаблон контекста «Обсуждения» предпочтительно (но не обязательно) содержит следующие параметры:

Минимальная длина потока для возврата. Пользователь факультативно указывает, что ему желательны потоки (цепочки) сообщений электронной почты по меньшей мере с одним ответом, с двумя ответами и т.д. Во многих случаях число потоков обеспечивает указание семантической важности. По умолчанию используется нулевое значение.

Фильтр списка распределения. Пользователь факультативно ограничивает возвращаемую электронную почту той, которая входит в состав одного или более списков распределения в строках "from", "to", "cc" или "bcc". Это позволяет пользователю контролировать дебаты от предпочтительных групп, отделений и т.д.

Фильтр строки распределения. Пользователь факультативно ограничивает возвращаемую электронную почту той, которая имеет фильтр адреса электронной почты в строках "from", "to", "cc" или "bcc". Возвращаемые элементы дополнительно сортируются на основе степени «свежести» или на основе глубины потока дебатов.

Шаблон контекста «Ньюсмейкеры». Шаблон контекста «Ньюсмейкеры» (и получаемый в результате специальный агент) может быть сопоставлен с персональной цифровой версией программы NBC «Встреча с прессой» в том, как он передает семантическую информацию. В этом случае акцент сделан на категории «люди в новостях», в противоположность самим новостям, или дебатам. Пользователи перемещаются в сети с использованием возвращенной информации о людях в качестве опорных пунктов информационного объекта. Шаблон контекста «Ньюсмейкеры» можно представить себе как шаблон контекста «Заголовки», предпочтительно с фильтрами типа объекта «люди» или «пользователи» и с предикатами «автором является», «возможно, автором является», «имеется в», «аннотировано», «эксперт по» и т.д. (предикаты, связывающие людей с информацией).

Устанавливаемый по умолчанию предикат «релевантно для» предпочтительно используется для покрытия всех уместных конкретных предикатов. Порядок сортировки релевантной информации, например, ньюсмейкеров, основан на порядке «новостей, которые они создают», например, заголовков. Помимо параметров шаблона контекста «Заголовки», шаблон контекста «Ньюсмейкеры» предпочтительно содержит следующие факультативные параметры:

Фильтр списка распределения. Пользователь факультативно ограничивает возвращаемую электронную почту той, которая входит в состав одного или более списков распределения в строках "from", "to", "cc" или "bcc". Это позволяет пользователю контролировать дебаты от предпочтительных групп, отделений и т.д.

Фильтр строки распределения. Пользователь факультативно ограничивает возвращаемую электрон-

ную почту той, которая имеет фильтр адреса электронной почты в строках "from", "to", "cc" или "bcc".

Шаблон контекста «Наступающие события». Шаблон контекста «Наступающие события» (и получаемый в результате специальный агент) может быть сопоставлен с персональной цифровой версией специальных программ, которые передают формацию о наступающих событиях. Примеры включают особенности событий, таких как «мировая серия», «финалы NBA», «финалы Кубка Мира по футболу» и т.д.

Эквивалентом в сценарии для специалиста в области информационных технологий является пользователь, который желает контролировать все наступающие события в отрасли, которые относятся к одной или нескольким категориям, документы или другие опорные пункты информационных объектов. Шаблон контекста «Наступающие события» предпочтительно идентичен шаблону контекста «Заголовки», за исключением того, что только наступающие события отфильтровываются и отображаются (предпочтительно с использованием соответствующей «поверхности контекста», которая ассоциирует события и критичность к времени). Возвращаемые объекты предпочтительно сортируются на основе критичности к времени, причем наиболее близкие события перечисляются первыми.

Шаблон контекста «Открытие». Шаблон контекста «Открытие» (и получаемый в результате специальный агент) может быть сопоставлен с персональной цифровой версией «канала «Открытие». В этом случае акцент сделан на «документации», относящейся к конкретным предметам. В отличие от случая «кратких новостей», главным стержнем для доступа и извлечения семантической информации является не время. Напротив, это одна или более категории с интеллектуальным агрегированием информации по этим категориям. В предпочтительном варианте осуществления настоящего изобретения шаблон контекста «Открытие» моделирует интеллектуальное агрегирование информации путем случайного выбора информационных объектов, которые связаны с данным набором категорий и которые посылаются за дополнительно предварительно определенный настраиваемый интервал времени. В то время как имеется факультативно настраиваемый интервал времени, семантический вес, в противоположность времени, является предпочтительным фактором учета для определения того, каким образом информация должна быть упорядочена или представлена. Настоящее и изобретение обеспечивает возможность использования различных осей, например, семантического веса для категории или категорий, которые «открывают», времени, случайности или комбинации всех осей (что, вероятно, должно повысить эффективность «открытия»). Шаблон контекста «Открытие» предпочтительно имеет те же параметры, что и шаблон контекста «Заголовки», за исключением того, что интервал времени «свежести» (информации) заменен на факультативный максимальный предельный возраст, что показывает максимальный возраст информации (посланной в агентство), которую должен возратить агент.

Шаблон контекста «предыстория». Шаблон контекста «предыстория» (и получаемый в результате специальный агент) может быть сопоставлен с персональной цифровой версией «исторического канала». В этом случае акцент сделан на распространении информации не о конкретных предметах, а в историческом контексте. Для данного шаблона предпочтительными осями являются категория и время. Шаблон контекста «предыстория» подобен шаблону контекста «открытие» дополнительно во взаимодействии с «минимальным предельным возрастом». Параметры предпочтительно те же самые, что и параметры шаблона контекста «открытие», за исключением того, что параметр «максимальный предельный возраст» заменен на «минимальный предельный возраст» (или дополнительный параметр «исторический временной отрезок»). Кроме того, возвращаемые объекты предпочтительно сортируются в обратном порядке, на основе их возраста в системе или их возраста с момента создания.

Шаблон контекста «все варианты выбора». Шаблон контекста «все варианты выбора» (и получаемый в результате специальный агент) представляет контекст, который возвращает любую информацию, которая релевантна на основе семантики или поиска по ключевым словам или на основе текста. В этом случае акцент сделан на распространении информации, которая может быть даже отдаленно релевантна контексту. Основной осью для шаблона контекста «все варианты выбора» предпочтительно является просто возможность релевантности. В предпочтительном варианте шаблон контекста «все варианты выбора» использует как семантический, так и основанный на тексте запрос для получения наиболее широкого возможного набора результатов, которые могут быть релевантными.

Шаблон контекста «наилучшие варианты выбора». Шаблон контекста «наилучшие варианты выбора» (и получаемый в результате специальный агент) представляет контекст, который возвращает наиболее релевантную информацию. В предпочтительном варианте акцент сделан на распространении информации, которая считается наиболее релевантной контексту и семантически значимой. Основной осью для данного шаблона контекста является релевантность. По существу, шаблон контекста «наилучшие варианты выбора» использует семантический запрос и не будет использовать основанные на тексте запросы, поскольку это не может гарантировать релевантность результатов запроса на основе текста. Шаблон контекста «наилучшие варианты выбора» предпочтительно инициализируется фильтром категории или ключевыми словами. Если ключевые слова определены, категоризация выполняется сервером динамически. Результаты предпочтительно сортируются на основе оценки релевантности или силы семантической связи «принадлежит к категории» от объекта к фильтру категории.

Шаблон контекста «предпочтения». Шаблон контекста «предпочтения» (и получаемый в результате

специальный агент) представляет контекст, который возвращает «любимую» или «популярную» информацию. В этом случае акцент сделан на распространении информации, которая рекомендована другими и принята в качестве предпочтительной. В предпочтительном варианте осуществления оси для шаблона контекста «предпочтения» включают уровень интереса читателей, полученные «представления» объекта и глубину потока аннотаций по данному объекту. В одном варианте осуществления шаблон контекста «предпочтения» возвращает только информацию, которая имеет семантические связи «предпочтений» и сортируется путем подсчета числа «голосов» за объект (на основе его семантической связи).

Шаблон контекста «образцы». Шаблон контекста «образцы» (и получаемый в результате специальный агент) представляет контекст, который возвращает «образцовую» информацию или информацию, значение которой признано. Подобно шаблону контекста «предпочтения», акцент сделан на распространении информации, которая рекомендована другими и принята в качестве предпочтительной. Для данного шаблона контекста предпочтительные оси включают в себя исторический контекст, уровень интереса читателей, полученные «представления» объекта и глубину потока аннотаций по данному объекту. Шаблон контекста «образцы» предпочтительно реализуется на основе шаблона контекста «предпочтения», но с дополнительным фильтром минимального предельного возраста, по существу, функционирующим как шаблон контекста «прежние предпочтения».

Шаблон контекста «рекомендации». Шаблон контекста «рекомендации» (и получаемый в результате специальный агент) представляет контекст, который возвращает «рекомендованную» информацию или информацию, которая на основе логических выводов агентств будет представлять интерес для пользователя. Рекомендации вводятся путем добавления семантических связей «рекомендации» к таблице «Семантические связи» и путем отыскания предпочтительных семантических связей, указанных пользователями. Рекомендации предпочтительно осуществляются с использованием таких методов, как машинное обучение и совместная фильтрация. Акцент данного шаблона контекста сделан на распространении информации, которая, по всей вероятности, будет представлять интерес для пользователя, но которую пользователь мог еще не видеть. Для данного шаблона контекста основные оси предпочтительно включают в себя вероятность интереса и свежесть (новизна). В предпочтительном варианте осуществления этот шаблон контекста реализуется путем генерации SQL, который имеет предикат PREDICATETYPEID_ISLIKELYTOBEINTERESTEDIN (ИД типа предиката «вероятно, заинтересован в») в качестве основного фильтра предиката для агентств в семантической среде.

Шаблон контекста «сегодня». Шаблон контекста «сегодня» (и получаемый в результате специальный агент) представляет контекст, который возвращает информацию посланную или сохраняемую (в случае событий) «сегодня». Акцент в этом шаблоне контекста сделан на распространении информации, которая оценивается как текущая на основе использования фильтра «сегодня» для определения свежести информации. В предпочтительном варианте результатом шаблона контекста «сегодня» является множество результатов шаблона контекста «заголовки», в котором отображаются результаты, посланные «сегодня», или события, сохраняемые «сегодня».

Шаблон контекста «разное». Шаблон контекста «разное» (и получаемый в результате специальный агент) представляет контекст, который возвращает случайную информацию. Акцент в этом шаблоне контекста сделан на распространении информации, которая является случайной, для того чтобы пользователь получил широкий диапазон возможных информационных объектов. В предпочтительном варианте основной осью является случайность, хотя «случайные» элементы будут семантически релевантны фильтру запроса (использующему предикат «релевантно для»).

b. Поверхности контекста.

Настоящее изобретение включает в себя специальный класс поверхностей, называемых «Поверхностями контекста». Поверхности контекста включают в себя информацию представления, которая передает семантику контекста, которую они представляют. Например, поверхность контекста для шаблона контекста «сегодня» может отображать эффекты фона или фильтра с часами, показывающими полночь, или некоторое иное представление «сегодня». В других примерах поверхность контекста для шаблона контекста «разное» может показывать эффекты трансформации, подобные шарам в боулинге, падающим случайным образом (указывая на случайность результата); поверхность контекста «новости, вызывающие прерывание», может показывать эффекты и анимации с мерцающим текстом, красные огни скорой помощи» и т.д., для указания на критический характер контекста; поверхность контекста «предыстория» может показывать графику, которая отображают «возраст» информации, например, старые автомобили, часы и т.д. Поверхности контекста предпочтительно сохраняют шаблон представления для типов объектов, которые отображаются. Например, объекты электронной почты могут отображаться на фоне, показывающем марки или почтовый фургон в дополнение к графике, отображающей шаблон контекста. Поскольку некоторые шаблоны контекста затрагивают несколько агентств - и поэтому затрагивают соответствующие онтологии - им не нужно отображать информацию, которая указывает на онтологию (например, информация из промышленности). Однако поверхности контекста, которые инициализированы фильтром категорий, предпочтительно показывают категорию или онтологию шаблона контекста. В типовом случае это будет представлено с графическими элементами (и фильтрами, трансформациями и т.д.), которые отображают отрасль промышленности или род онтологии. Например, поверхность контекста

ста «фармацевтика» может использовать эффекты фильтра, показывающие лабораторное оборудование; поверхность контекста «нефть и газ» может показывать картинки буровых установок; поверхность контекста «спорт» может показывать картинки спортивного снаряжения и т.д.

с. Шаблоны поверхностей.

Настоящее изобретение обеспечивает возможность пользователям выбирать различные виды поверхностей, в зависимости от конкретной решаемой задачи. Предпосылкой использования гибкого представления является то, что пользователь может выбрать наилучший режим представления на основе текущей задачи. Например, пользователи могут выбрать утонченную (незаметную) поверхность, когда они работают на своей основной машине и когда производительность является наиболее критичным фактором, а эффекта нет. Пользователи могут выбрать поверхности умеренного характера в случаях, когда производительность также важна, но и эффекты также было бы желательно отображать. Пользователи также могут выбрать поверхности захватывающего характера для сценариев подобной работе с второстепенными машинами, например, когда пользователи просматривают информацию периферийным зрением и важными являются свойства, выраженные через текст и речь, чтобы оповестить их о критически важных новостях. Такие возбуждающие внимание поверхности могут отличаться анимацией, эффектами подобными архивным отображениям для глубокого представления информации, объектами, отображаемыми на траектории движения, и другими эффектами. Возбуждающие внимание поверхности особенно пригодны для использования с хранителями экрана. Выбор поверхностей предпочтительно определяется пользователем.

d. Устанавливаемые по умолчанию предикаты.

В предпочтительном варианте осуществления каждый тип объекта включает в себя устанавливаемый по умолчанию предикат, который связывает его с другими типами объектов. Это обеспечивает пользователю интуитивным методом динамического связывания объектов вместе, не требуя отдельной оценки предиката для использования для семантической связи. Например, операция «перетащить и оставить» из объекта «документ» к агенту, который возвращает документ, может иметь предикаты «относится к», «возможно, относится к». Если объект «документ» перемещается поверх агента «документ», семантический браузер настоящего изобретения отображает опцию всплывающего меню, которая позволяет пользователю выбрать предикат для использования для семантического запроса. В альтернативном варианте осуществления могут быть включены другие связанные всплывающие меню, например, первое всплывающее меню, которое позволяет пользователю выбрать связь или шаблон предиката; дочерние всплывающие меню, которые отображают действительные предикаты для выбранного шаблона. Устанавливаемый по умолчанию предикат предпочтительно вводится в динамически генерируемый SQL, из которого будет активизироваться запрос. Например, устанавливаемый по умолчанию предикат может быть «относится к». Этот предикат отображает на запрос ту возвращаемую информацию в агенте «документ», которая релевантна объекту, который перемещается. Преимущество наличия устанавливаемого по умолчанию предиката в этом случае состоит в том, что семантический браузер настоящего изобретения может отображать опцию всплывающего меню «открыть», которая в свою очередь активизирует запрос с использованием этого предиката. Семантический браузер может также отобразить опцию всплывающего меню «открыть со связью», которая имеет опции подменю с конкретными предикатами. Устанавливаемый по умолчанию предикат обеспечивает простоту использования системы, поскольку пользователи могут просмотреть ресурсы системы с использованием динамического связывания, зная, что устанавливаемый по умолчанию предикат будет практичной опцией, выдающей исходный объект и агента или объект, являющийся целевым.

В дополнение к использованию в сценариях «перетащить и оставить» устанавливаемые по умолчанию предикаты факультативно используются в интеллектуальных лупах, интеллектуальных операциях «копировать и вставить» и т.д. Устанавливаемые по умолчанию предикаты могут быть сопоставлены с вырожденными интеллектуальными связями, которые возвращают «правильную вещь» в качестве соответствующего результата запроса для семантического расстояния «единица». В альтернативном варианте устанавливаемый по умолчанию предикат может являться объединением различных конкретных предикатов. Например, устанавливаемый по умолчанию предикат для операций «перетащить и оставить», «копировать и вставить» и интеллектуальной лупы с использованием объектов «документ-люди» может представлять собой «релевантно для» и может интерпретироваться XML-Web-сервисом агентства KISD как, например, каскадно связанный запрос, включающий предикаты «автором является», «эксперт по», «аннотировано». Иными словами, в соответствии с настоящим изобретением, «релевантность» интерпретируется интеллектуальным образом и может предусматривать объединение различных предикатов.

Устанавливаемые по умолчанию предикаты позволяют пользователям быстро и эффективно перемещаться в системе, по существу, не задумываясь. Устанавливаемые по умолчанию предикаты придают системе простоту и обеспечивают возможность интуитивного ее использования. Кроме того, пользователям удобно использовать устанавливаемые по умолчанию предикаты, поскольку они уже использовали для активизации HTML-связей в современной сети Web имеющийся в ней предикат «активизировать».

e. Предикаты контекста.

Предикаты контекста являются предикатами, которые определены на высоком уровне абстракции и

которые отображаются на релевантное подмножество шаблонов контекста. Предикаты контекста позволяют пользователям выбрать фильтр предиката на основе шаблона контекста, а не на основе системного предиката низкого уровня. Когда запрос активизируется с использованием предиката контекста, фильтрация SQLML содержимого с параметрами фильтра шаблона контекста генерирует новый SQLML-запрос. Например, предикат контекста «наилучшие варианты выбора» отображается на шаблон контекста того же имени и фильтрует запрос с теми информационными объектами, которые являются «наилучшими вариантами выбора» (в типовом случае они будут элементами, которые возвращены исходя из семантического запроса, а не из запроса на основе текста). Аналогичным образом, предикат контекста «новости, вызывающие прерывание» фильтрует элементы на основе того, определены ли они условиями фильтра шаблона контекста «новости, вызывающие прерывание». В принципе, предикаты контекста применимы для типов объектов, которые согласованы с шаблоном контекста (например, предикаты контекста «эксперты» и «нюсмейкеры» будут действительны только для запросов, которые возвращают объекты «персона»).

f. Атрибуты контекста.

Атрибуты контекста являются «виртуальными атрибутами», которые кэшированы как часть каждого XML-объекта, который агентство возвращает клиенту. Эти атрибуты являются динамическими в том, что они отражают текущий контекст, в котором отображаются результаты. Например, где релевантно, атрибут контекста «наилучший вариант выбора» присоединяется к каждому XML-результату, который удовлетворяет фильтру семантического запроса в SQLML текущего запроса. Результаты семантического запроса с устанавливаемыми по умолчанию предикатами могут включать в себя как семантические, так и несемантические (на запросы на основе текста) результаты. Агентство, обрабатывающее запрос, может кэшировать атрибуты контекста для XML-результатов, которые являются «наилучшими вариантами выбора», путем исполнения семантического подзапроса на SQLML с результирующим объектом в качестве фильтра. В этом случае схемы для «объекта» и производных типов должны включать в себя поля атрибутов для каждого релевантного шаблона контекста (например, атрибута «наилучший вариант выбора», атрибута «заголовок» и т.д.). Это является предпочтительной реализацией. Альтернативно семантический браузер обращается к агентству, передает каждый XML-объект как аргумент и «спрашивает», удовлетворяет ли объект атрибуту контекста. Другими примерами являются атрибут контекста «заголовок», который указывает, определяется ли объект как «заголовок» в контексте текущего запроса, атрибут «образцы» и т.д. Семантический браузер должен отображать пользовательский интерфейс, указывающий, установлен ли атрибут контекста или нет.

Атрибуты контекста обеспечивают дополнительные преимущества по сравнению с известными системами, заключающиеся в обеспечении большей простоты использования системы. Например, пользователь может выполнить операцию «перетащить и оставить» для генерации реляционного запроса, который включает в себя как семантические, так и несемантические фильтры запроса (как обработано агентством, когда оно принимает SQLML-аргументы от клиента). В одном варианте браузер «спрашивает» пользователя, желателен ли ему расширенный запрос или запрос «наилучших вариантов выбора». В этом режиме пользователь эффективно применяет дополнительный фильтр перед выдачей запроса. Альтернативно, агентство совместно с семантическим браузером предпочтительно возвращает результаты расширенного запроса, а также определяет каждый результат атрибутом контекста и соответствующим пользовательским интерфейсом, указывающим, является ли каждый результирующий объект «расширенным» или «наилучшим вариантом выбора». То же самое применимо к другим типам объектов, подобным типу объекта «персона». Вместо того, чтобы пользователь указывал, должен ли реляционный запрос к агенту «персона» возвращать «авторов», «экспертов» или «референтов», браузер может выдать расширенный запрос и затем определить результаты (с помощью от агентства) тем, является ли каждая возвращенная «персона» «автором», «экспертом» или «референтом» для текущего контекста.

g. Палитры контекста.

Палитры контекста являются очень эффективным свойством настоящего изобретения, которое связано с динамической активизацией шаблонов контекста для текущего выбранного объекта в рамках семантического браузера. По существу, палитры контекста предпочтительно вызываются автоматически и отображаются, когда пользователи выбирают какой-либо объект в панели результатов. Палитры контекста дают возможность пользователям всегда иметь в распоряжении контекст для текущих отображаемых результатов. Кроме того, семантический браузер постоянно обновляет палитру для текущего выбранного объекта, тем самым гарантируя то, что контекст для объекта всегда является обновленным. В предпочтительном варианте это выполняется с помощью таймера, который запускает действие обновления или путем запроса процессора SQLML-запросов о палитре контекста, имеется ли какой-либо новый объект с момента последнего обновления палитры.

В предпочтительном варианте результаты, отображаемые в палитре контекста, являются «первоклассными» информационными объектами тем же путем, что и информационные объекты, отображаемые в основной панели результатов. Иными словами, результаты палитры контекста предпочтительно используются со всеми семантическими инструментальными средствами настоящего изобретения, например, интеллектуального копирования и вставки, интеллектуальной лупы, глубинной информации и

т.д. То же самое предпочтительно верно для результатов, отображаемых в других панелях контекста согласно настоящему изобретению.

Настоящее изобретение предпочтительно включает в себя следующие палитры контекста. В предпочтительном варианте пользователи имеют опцию «прокрутки» различных палитр контекста для выбранного объекта. Включение дополнительных или отличающихся палитр контекста является очевидным и может быть параллельным с пополнением шаблонов контекста.

Палитра контекста «заголовки». Это использует шаблон контекста «заголовки» и применяет SQML, имеющий SQML шаблона контекста «заголовки» с дополнительными связями к текущему выбранному объекту, и устанавливаемый по умолчанию предикат для комбинации «объект-тип». В частности, SQML будет вводиться из ресурсов, которые отображаются на все предпочтительные агенты или последние агенты в семантической среде. Пользователь конфигурирует, желательны ли ему предпочтительные агенты, последние агенты, или и то, и другое, для использования при генерировании палитры контекста. Кроме того, палитра контекста «заголовки» также может настраиваться для показа заголовков без какой-либо фильтрации для ряда объектов, которые должны отображаться, или предела времени «обновления». В этом случае палитра позволит пользователю перемещаться по всем реляционным результатам, отсортированным по времени публикации или посылки.

Палитра контекста «новости, вызывающие прерывание». Содержит реляционные результаты из каждого агента «новости, вызывающие прерывание», в семантической среде с использованием устанавливаемого по умолчанию предиката комбинации «объект-тип» и в связи с текущим выбранным объектом. Кроме того, отображаются результаты из устанавливаемой по умолчанию палитры контекста «новости, вызывающие прерывание». Семантический браузер настоящего изобретения будет динамически генерировать SXQML с таким количеством (идентичных) ресурсов или комбинаций связей, как если бы имелись агенты «новости, вызывающие прерывание», с дополнительными связями, которые имеют устанавливаемые по умолчанию предикаты, и спецификаторы ресурсов текущего выбранного объекта (путь к файлу, путь к папке, объект://URL и т.д.). Семантический браузер настоящего изобретения активизирует генерированный SQML-запрос и загружает окна палитры с SQML-результатами. Палитра контекста «новости, вызывающие прерывание», предпочтительно содержит элементы управления навигацией, чтобы обеспечить возможность пользователям перемещаться по результатам в палитре контекста.

Палитра контекста «обсуждения». Подобна палитре контекста «заголовки», за исключением использования шаблона контекста «обсуждения».

Палитра контекста «нюсмейкеры». Подобна палитре контекста «заголовки», за исключением использования шаблона контекста «нюсмейкеры».

Палитра контекста «наступающие события». Подобна палитре контекста «заголовки», за исключением использования шаблона контекста «наступающие события».

Палитра контекста «открытие». Подобна палитре контекста «заголовки», за исключением использования шаблона контекста «открытие».

Палитра контекста «предыстория». Подобна палитре контекста «заголовки», за исключением использования шаблона контекста «предыстория».

Палитра контекста «все варианты выбора». Подобна палитре контекста «заголовки», за исключением использования шаблона контекста «все варианты выбора».

Палитра контекста «наилучшие варианты выбора». Подобна палитре контекста «заголовки», за исключением использования шаблона контекста «наилучшие варианты выбора».

Палитра контекста «предпочтения». Подобна палитре контекста «заголовки», за исключением использования шаблона контекста «предпочтения».

Палитра контекста «образцы». Подобна палитре контекста «заголовки», за исключением использования шаблона контекста «образцы».

Палитра контекста «рекомендации». Подобна палитре контекста «заголовки», за исключением использования шаблона контекста «рекомендации».

Палитра контекста «сегодня». Подобна палитре контекста «заголовки», за исключением использования шаблона контекста «сегодня».

Палитра контекста «разное». Подобна палитре контекста «заголовки», за исключением использования шаблона контекста «разное».

Палитра контекста «временная шкала». Эта палитра контекста предпочтительно содержит объединенные результаты из шаблонов контекстов «заголовки», «наилучшие варианты выбора», «предыстория» и «наступающие события». Палитра контекста «временная шкала» предпочтительно позволяет пользователю перемещаться по всем объектам на семантической временной шкале на основе текущего выбранного объекта. Временная шкала может содержать информационные элементы, основанные на их времени публикации/посылки, элементы событий, основанные на их назначенном времени, и т.д. По существу, с помощью палитры контекста «временная шкала» пользователь перемещается по релевантным объектам (и, возможно, другим семантически связанным объектам) с использованием времени в качестве основной оси для доставки информации.

Палитра контекста «проводник». Предпочтительный вариант осуществления настоящего изобре-

ния включает в себя объединенную палитру контекста «проводник». Эта палитра контекста объединяет все палитры контекста. Иными словами, каждое окно в палитре контекста «проводник» соответствует одному результату из каждой из других палитр контекста системы. Пользовательский интерфейс для палитры контекста «проводник» позволяет пользователям осуществить прокрутку результатов по каждой палитре контекста в каждом окне или анимировать результаты с использованием методов анимации, например, методов плавного введения/удаления изображения. Предпочтительное использование палитры контекста «проводник» состоит в просмотре контекста для текущего выбранного объекта в минимальном пространстве представления. В предпочтительном варианте осуществления такое использование имеет опцию просмотра всех палитр контекста рядом (по вертикали, по горизонтали, по диагонали и т.д.), пристыкованными или в другом формате упорядочивания.

Пользовательский интерфейс палитры контекста.

Пользовательский интерфейс для палитр контекста предпочтительно может конфигурироваться на основе топологической поверхности для текущего отображаемого агента. В предпочтительном варианте палитры контекста могут быть пристыкованы слева, справа, сверху или снизу к панели результатов. Палитры контекста могут сворачиваться для минимизации внедрения в область отображения и динамически разворачиваться до полного представления. Поверхности могут обеспечить возможность палитре контекста изменить свои размеры на переменные или предварительно установленные размеры. Альтернативно, некоторые поверхности могут также анимировать результаты палитр контекста.

К примеру, фиг. 80 иллюстрирует пользовательский интерфейс, представляющий результаты агента и соответствующие палитры контекста. В этом примере некоторые палитры контекста свернуты, причем палитры контекста представлены как вертикально пристыкованные с правой стороны дисплея или панели результатов.

h. Внутренние оповещения.

В предпочтительном варианте осуществления в дополнение к агенту «новости, вызывающие прерывание», настоящее изобретение предусматривает внутренние оповещения. Будучи концептуально подобными агентам новостей, вызывающих прерывание, внутренние оповещения фундаментальным образом отличаются по своему действию. В случае агентов новостей, вызывающих прерывание, настоящее изобретение предусматривает сигнализацию пользователю, что касается новостей, вызывающих прерывание, после опроса каждого агента новостей, вызывающих прерывание, определенного пользователем, и запроса у него о поиске чего-либо связанного с текущим объектом, вызывающим прерывание. Внутреннее оповещение не требует от пользователя конкретно определять агента новостей, вызывающих прерывание, или иным образом выполнять какое-либо действие для введения уведомления о новостях, вызывающих прерывание. Внутреннее оповещение осуществляет автоматическую сигнализацию в пользовательском интерфейсе (для всех текущих отображаемых объектов), когда имеется событие, которое относится к данному объекту, путем выдачи внутренне присущим, свойственным ему способом. Например, если текущий объект является документом, настоящее изобретение предусматривает опрос агентства, из которого получен документ, имеется ли на агентстве какая-либо недавно полученная информация, относящаяся к данному объекту. Если текущий объект является персоной, агентство может опрашиваться о том, послало ли это лицо недавно электронную почту, послало ли оно недавно документ, аннотировало ли оно недавно документ, присоединенный или выведенный из списка распространения, и т.д. Это позволяет пользователю иметь непосредственно «на месте» информацию в рамках собственного контекста объекта зависящим от времени способом.

В предпочтительном варианте устанавливаемая по умолчанию реализация для внутренних оповещений будет опрашивать только агентство, откуда поступил объект. Преимуществом этого является упрощение пользовательского интерфейса; если пользователю желательно выполнить перекрестные запросы между агентствами, он может воспользоваться опцией «перетащить и оставить», «копировать и вставить» и т.д., чтобы активизировать реляционные запросы. В альтернативном варианте внутренние оповещения будут опрашивать множество агентств, включая агентства иные, чем те, из которых поступил объект, чтобы локализовать уведомления о новостях, вызывающих прерывание.

В альтернативном варианте осуществления настоящее изобретение предусматривает настройку для поддержания информации о том, получил ли пользователь доступ к объекту. Это может быть сопоставлено с тем, как сервер электронной почты отслеживает сообщения электронной почты, которые прочитал пользователь. В возможном варианте, в котором агентство поддерживает состояние серверной стороны по каждому объекту, по каждому пользователю, внутренние оповещения всегда являются точными, так как агентство указывает, что имеются «новости, вызывающие прерывание», только если в агентстве имеется информация, которая связана с рассматриваемым объектом, к которому пользователь не обращался или который не был прочитан пользователем. Эта альтернатива предпочтительно выполняется посредством дополнительной фильтрации SQL-запроса.

Этот вариант, использующий состояние серверной стороны по каждому объекту, по каждому пользователю, имеют недостатки, особенно для агентств, которые должны поддерживать большие массивы информации и иметь огромное количество пользователей (например, агентства на базе Интернет). В этой ситуации система не масштабируется надлежащим образом, если состояние поддерживается по каждому

объекту и по каждому пользователю.

В альтернативном варианте осуществления, в котором агентство не поддерживает состояние по каждому объекту и по каждому пользователю, агентство может конфигурироваться со статическим пределом времени обновления для внутренних оповещений. Например, сервер может конфигурироваться со статическим пределом времени обновления, равным тридцати минутам, и в этом случае сервер будет реагировать подтверждением, если запрос внутреннего оповещения принят в течение 30 мин от поступления нового объекта, который относится к объекту в запросе. В предпочтительном варианте осуществления агентство KIS поддерживает информацию о средней скорости поступления информации. Таким путем загруженный сервер будет иметь более низкий предел времени обновления, чем сервер, который редко принимает новую информацию. Этот вариант осуществления не так точен, как тот, когда сервер поддерживает состояние по каждому объекту и по каждому пользователю, поскольку средняя скорость поступления обеспечивает лишь аппроксимацию того, следует ли сигнализировать уведомлением. Данный вариант приводит к пониженным потерям информации. В предпочтительном варианте настоящего изобретения предусматривается факультативную сигнализацию посредством внутренних оповещений неинтрузивным способом, который поддерживает вероятностную природу оповещений (т.е. оповещение является только наилучшим предположением).

i. Интеллектуальные рекомендации.

Интеллектуальные рекомендации представляют семантические запросы в семантическую сеть для логически выведенных семантических связей, с использованием объекта в качестве опорного пункта информационного объекта. Например, механизм логического вывода может логически вывести, что пользователи хотели бы посетить определенное событие, основываясь на событиях, которые они посещали в прошлом, на том факте, что они принимали участие в множестве диалогов по электронной почте с презентатором события и т.д. Например, в предпочтительном варианте осуществления эта информация доступна в панели результатов контекста интеллектуальных рекомендаций, как показано на фиг. 81. Это подобно тому, что пользователи наблюдают для данного объекта в отношении шаблона контекста «рекомендации».

В предпочтительном варианте осуществления каждая связь генерируется поверхностью объекта или специальной поверхностью информационной панели рекомендаций и будет связана с SQML, содержащим предикаты для логически выведенных семантических связей.

6. Преимущества свойств настоящего изобретения.

Информационная нервная система согласно настоящему изобретению обеспечивает надлежащий контекст, значение (смысл) и эффективный доступ к данным и информации для обеспечения возможности пользователям приобретать действенные знания. Многие из преимуществ информационной нервной системы по сравнению с современной сетью Web и концептуальной семантической сетью Web вытекают из использования в ней уровней технологии, показанных на фиг. 82. Различные варианты осуществления настоящего изобретения демонстрируют преимущества, связанные со свойствами, требуемыми для того, чтобы создать интегрированную и непрерывную реализацию базовой структуры и результирующей среды для извлечения, управления и доставки знаний, которые включают в себя семантику/значение; зависимость от контекста; зависимость от времени; автоматическую и интеллектуальную обнаружимость; динамическое связывание; управляемые пользователем навигацию и просмотр ресурсов; участие в информационном обмене в сети не-HTML и локальных документов; гибкое представление, которое интеллектуальным образом передает семантику отображаемой информации; логику, логический вывод и умозаключение; гибкий, управляемый пользователем анализ информации; гибкие семантические запросы; считывание/запись в сети Web; аннотации; «сеть Web доверия»; информационные пакеты («элементы сопряжения»); шаблоны контекста; ориентированное на пользователя агрегирование информации.

Семантика/Значение.

Настоящее изобретение использует семантические связи, онтологии и другие хорошо определенные модели данных, использующие XML. В результате агентство, как описано выше, имеет возможности семантического Web-сайта в том, что его информация включает в себя семантику. Кроме того, за счет обеспечения значения как внутренне присущей (собственной) части XML-Web-сервера, оно также обеспечивает зависимость от контекста, времени и т.д., связанную с информацией предметной области. Зависимые от контекста агенты интеллектуальной системы, описанные выше, контролируют собственный контекст пользователей и автоматически оповещают пользователей, когда появляется релевантная информация в источнике (или источниках) информации, относящаяся к определенному контексту. К примеру, такие определенные контексты могут включать в себя следующее:

Мои документы
Мой Web-портал
Мои любимые Web-сайты
Моя электронная почта
Мои контакты
Мой календарь
Мои клиенты

Моя музыка
 Мое местоположение
 «Этот» документ
 «Эти» Web-сайт/страница
 «Это» сообщение электронной почты
 «Этот» контакт
 «Это» событие в моем календаре
 «Этот» клиент
 «Эта» музыкальная дорожка, альбом, список для воспроизведения.

Настоящее изобретение обеспечивает контекстно-зависимый пользовательский опыт посредством использования информационных агентов, связанных с сервером 10, и посредством семантического браузера 30, связанного с XML-Web-сервисом. Например, пользователи автоматически связывают информацию в «моих документах», «моей электронной почте» и т.д. (из «островков» приложений, таких как Microsoft Outlook и т.д.) с удаленными источниками информации, которые содержат семантически релевантную информацию. Пользователи обладают гибкостью для реализации этих соединений в реальном времени на основе инноваций на уровне приложений, которые помещаются поверх семантической сети, таких как новые инструментальные средства запросов, описанные выше, например, операция «перетащить и оставить», интеллектуальная лупа, интеллектуальное «копирование и вставка» и т.д. Также предусматривается, что инструментальные средства приложений могут быть использованы независимо от семантической сети, например, интегрированными в существующий браузер современной сети Web.

В предпочтительном варианте KIS настоящего изобретения выводит семантическую информацию из семантической сети Web или другого хранилища с семантической разметкой (предпочтительно посредством сменных модулей стандарта RDF) в свою семантическую сеть. Альтернативно, система 10 настоящего изобретения существует без семантической сети Web. В этой ситуации KIS строит свою собственную семантическую сеть (например, частную семантическую сеть) из информационных источников, которые выбирает администратор системы (например, электронная почта, документ и т.д.). Система 10 настоящего изобретения способна использовать актуальные семантические приложения с семантической машиной базы данных (что может факультативно включать в себя семантическую Web-сеть). Система 10, таким образом, обеспечивает зависимость от контекста посредством интеграции с клиентскими приложениями (включая специализированный (собственный) семантический браузер 30), инструментальными средствами отслеживания местоположения и т.д. и специализированным XML-Web-сервисом (который семантическая сеть Web не описывает). Более конкретно, хотя концептуальная семантическая сеть Web описывает архитектуру для семантического связывания и представления знаний, она не рассматривает сценарии и инновации, использующие XML-Web-сервисы для обеспечения зависимости от контекста, зависимости от времени, динамического связывания, шаблонов контекста, палитр контекста и т.д. В противоположность этому, настоящее изобретение рассматривает семантическое связывание посредством модели семантических данных и семантической сети, а также обеспечивает программные сервисы для обеспечения зависимости от контекста, зависимости от времени, динамического связывания, шаблонов контекста, палитр контекста и т.д. посредством интеграции со специализированным XML-Web-сервисом.

Временная зависимость.

Настоящее изобретение имеет внутренне присущее представление временной зависимости. Например, за счет обеспечения признаков, связанных с временной зависимостью, таких как агенты новостей, вызывающих прерывание, шаблонов контекста новостей, вызывающих прерывание, палитра контекста новостей, вызывающих прерывание, и соответствующих внутренних оповещений, настоящее изобретение демонстрирует важность времени как элемента в семантике и представлении.

Хотя и абсолютно истинно, однако, в принципе, старая информация не настолько релевантна, как новая информация. Например, CNN прерывает новостное вещание для показа срочных новостей, при этом прерывание основывается на комбинации семантики (релевантности этих срочных новостей относительно того, что должно отображаться) и того факта, что новости на самом деле являются крайне важными. За исключением тех редких случаев, когда Web-автор специально выстраивает анализ с временным приоритетом, элемент зависимости от времени в качестве «оси» для оповещений и представления полностью отсутствует в современной сети Web и в концептуальной семантической сети Web.

Настоящее изобретение позволяет пользователям выбрать интеллектуальных агентов в качестве агентов новостей, вызывающих прерывание. Любая отображаемая информация будет показывать оповещения, если имеются релевантные срочные новости в агенте новостей, вызывающих прерывание. Например, согласно настоящему изобретению, пользователь может создать агента «все документы, посланные Reuters сегодня» или «все события, относящиеся к компьютерной технологии и сохраненные в Сиэтле в следующие 24 ч», в качестве агентов новостей, вызывающих прерывание. Поскольку эти агенты являются персональными («прерывание» (срочность информации) является субъективной и зависит от пользователя), браузер обеспечивает уникальную индивидуальную поддержку. В другом примере пользователь в Сиэтле хотел бы иметь возможность запланировать уведомление о событиях в Сиэтле в сле-

дующие 24 ч, о событиях на Западном побережье на следующей неделе (за это время пользователь может найти для себя недорогой авиационный рейс), о событиях в США в течение следующих 14 дней (предварительное уведомление для большинства американских авиаперевозчиков для получения билета на рейс, совершающий континентальный перелет, по конкурентоспособной цене), о событиях в Европе в следующем месяце (возможно, поскольку пользователю необходимо время для резервирования места в гостинице) и о событиях в мире в течение следующих шести месяцев.

Настоящее изобретение также поддерживает шаблон контекста новостей, вызывающих прерывание, на основе которого пользователи могут создавать агентов новостей, вызывающих прерывание. Также настоящее изобретение поддерживает палитру новостей, вызывающих прерывание, что позволяет пользователям просматривать все отображенные результаты в контексте определения на основе шаблона «новостей, вызывающих прерывание», при этом интегрируя непрерывно и интеллектуальным образом контекстную и временную зависимость.

Настоящее изобретение также обеспечивает мощное персональное архивное инструментальное средство для выполнения анализа предыстории. Использование просмотра предыстории, прошлых событий и времени создания документов, система 10 может компенсировать недостатки памяти путем вызова детальных данных из какого-либо события, например, показывая результаты запроса «сотрудники, которые присутствовали на обсуждениях проекта с 6/1/98 по 6/1/99». Альтернативно, система может отслеживать группы событий. Например, исследователи могут запросить «все рыночные сделки с ценными бумагами, превышающими \$10 М, относящиеся к ценным бумагам авиалиний, с 7/1/98 по 9/11/01» или «показать все документы, созданные в течение интервала времени в десять дней для этого события».

Интеллектуальная обнаружимость.

Система 10 настоящего изобретения имеет собственное представление об обнаружимости. В предпочтительном варианте осуществления KIS автоматически уведомляет о своем присутствии по локальной многоадресной сети, в каталоге предприятия (например, каталоге LDAP или активном каталоге Windows 2000), в одноранговой системе или иной системе. В идеальном случае, семантический браузер 30 периодически прослушивает многоадресные передачи или одноранговые оповещения и проверяет каталог предприятия или глобальный каталог агентств. Браузер также позволяет пользователям перемещаться в системе иерархическим способом для определения местонахождения дополнительных агентств. Таким путем пользователи уведомляются о том, когда становятся доступными новые агентства и когда действие существующих агентств прекращается. Семантический браузер настоящего изобретения предпочтительно мгновенно уведомляет пользователей, когда становятся доступными новые агентства, посредством моментальных отображений в пространстве имен и периодических проверок уведомлений и присутствия в каталоге.

Аспект одноранговости позволяет системе 10 масштабировать и автоматически заполнять каталог предприятия без централизованной поддержки (что влечет за собой большие текущие издержки для организаций). Система предпочтительно использует программируемые запросы для новых классов серверов, тем самым исключая необходимость в регистрациях в Web-сети.

Динамическое связывание.

Заявленная система 10 обеспечивает фундаментальные преимущества по сравнению с современной сетью Web и с концептуальной семантической сетью за счет использования интеллектуальных объектов, обладающих внутренне присущей им линией поведения. Система помещает такие поведенческие характеристики в каждый XML-Web-сервис агентства, тем самым делая каждый узел в семантической сети более интеллектуальным, чем обычная связь или узел в современной сети Web или семантической сети. Иными словами, в предпочтительном варианте осуществления каждый узел в семантической сети настоящего изобретения связывается с другими узлами независимо от авторства (авторской разработки). Каждый узел имеет линию поведения, которая динамически связывает с агентствами. Интеллектуальные агентства, таким образом, позволяют реализовывать такие дополнительные функции, как «перетащить и оставить», интеллектуальное копирование и вставка, создание связей с агентствами в семантической среде, реагирование на запросы из интеллектуальных агентств для создания новых связей, включение внутренних оповещений, которые будут динамически создавать связи с информацией, характеризующей временной зависимостью, на своем агентстве, включение указаний на новости, вызывающие прерывание (причем узел может автоматически связываться с агентами новостей, вызывающих прерывание, в пространстве имен), и т.д. Эти функции существенным образом расширяют возможности пользователей, например, по поиску и просмотру новых связей. Как только пользователь достигает узла в сети, он получает множество семантических средств динамической и автоматической навигации с использованием контекста, времени, связанности с интеллектуальными агентствами и агентами и т.д. Делая каждый узел в сети более интеллектуальным, вся семантическая сеть становится интеллектуальной, виртуальной, самоподдерживающейся и самосоздаваемой сетью.

Технология динамического связывания настоящего изобретения позволяет пользователям выдавать запросы через локальные/удаленные информационные границы. Например, настоящее изобретение (предпочтительно с использованием SQLML-технологии) позволяет пользователю выдать запрос такого рода: «Найти мне все сообщения электронной почты, написанные моим начальником или любым лицом

из исследовательского подразделения, которые относятся к данной спецификации на моем жестком диске». Технология обработки запросов на клиентской стороне (предпочтительно посредством SQL) обеспечивает возможность реализации такого гибкого запроса, поскольку процессор связывает метаданные от клиента с удаленным XML-Web-сервисом, который обрабатывает реляционный запрос.

Интеллектуальное и динамическое распространение информации. Динамическое связывание, предусмотренное в настоящем изобретении, обеспечивает интеллектуальное распространение информации. Поскольку семантическая сеть допускает навигацию в ней по намного большему количеству осей, чем современная сеть Web или семантическая сеть Web, то совместное использование информации и ее распространение становится намного более эффективным, и потери информации минимизируются.

Управляемое пользователем перемещение и просмотр.

Свойство динамического связывания настоящего изобретения обеспечивает возможность непрерывного семантического просмотра в противоположность современной сети Web или семантической сети Web, где статические связи приводят к просмотру «тупиковых» элементов данных. В современной сети Web или семантической сети Web пользователь в типовом случае просматривает ресурсы до желательного местоположения или по существу достигает тупика, где дальше не имеется никаких связей. При динамическом связывании пользователь может, в зависимости от свойства информационного пространства в данный момент времени, продолжать просмотр ресурсов бесконечно, поскольку узел сам включает в себя интеллектуальные свойства для динамического обновления связей.

Например, за счет непрерывной интеграции и связывания семантических XML-Web-сервисов, предусмотренных настоящим изобретением, пользователи «перетаскивают и оставляют» файлы, связи и т.д. в интеллектуального агента для создания новых интеллектуальных агентов. Предпочтительно это происходит рекурсивным образом. Интеллектуальные агенты, в свою очередь, могут быть преобразованы, при необходимости, в интеллектуальных агентов новостей, вызывающих прерывание. Другие узлы в представлении отображают указки представления, показывающие наличие срочных новостей на любом агенте новостей, вызывающих прерывание. В продолжение данного примера, результаты запроса агенту новостей, вызывающих прерывание, могут быть использованы как интеллектуальная лупа, что показывает дальнейшие результаты. Эти результаты предпочтительно включают в себя внутренние оповещения, которые обеспечивают пользователя контекстно- и времязависимым путем в сети.

Последующие результаты могут копироваться и вставляться в любое агентство, а также перетаскиваться и оставляться в других интеллектуальных агентах.

В предпочтительном варианте осуществления динамическое связывание настоящего изобретения применяется к объектам в семантической «песочнице» (среде временной изоляции загружаемого кода) (объектам, которые находятся в среде системы 10 и отображаются в семантическом браузере 30), а также к внешним объектам, которые могут динамически добавляться к среде. Это обеспечивает непрерывный динамический путь перемещения от существующих документов (в файловой системе, современной сети Web или в других средах) в систему 10, соответствующую настоящему изобретению.

На фиг. 83 иллюстрируется динамическое связывание и управляемые пользователем навигация и просмотр ресурсов в соответствии с предпочтительным вариантом осуществления настоящего изобретения. В целях данного примера, термин «Интеллектуальные связи» относится к динамической, программируемой семантической связи согласно настоящему изобретению.

Участие в сети не-HTML и локальных документов.

Настоящее изобретение не требует, чтобы документы были кодированы как RDF или XML перед их включением в сеть. Напротив, KIS (или сервер агентства) автоматически извлекает метаданные из всех типов объектов и добавляет их в семантическую сеть. Кроме того, динамическое связывание на клиентской стороне, предпочтительно с использованием таких инструментов, как «перетащить и оставить», интеллектуальное копирование и вставка и интеллектуальная лупа, гарантирует, что локальные документы всех типов связываются в сеть, тем самым увеличивая значимость и объем сети. Настоящее изобретение автоматически извлекает метаданные из локальных документов и обращается в KIS (посредством его XML-Web-сервиса) для извлечения семантически связанной информации. Таким образом, локальные документы не исключаются из сети. Настоящее изобретение дает возможность пользователю перетаскивать и оставлять документ из неинтеллектуальной среды (например, из современной сети Web или файловой системы) в систему 10, тем самым обеспечивая ее интеллектуальность. Как только метаданные оказываются в системе 10, семантические инструментальные средства, такие как семантические «лупы», интеллектуальное копирование и вставка и т.д. могут быть выполнены над объектами и с использованием объектов. Операция «перетащить и оставить» также поддерживается непосредственно из пользовательской файловой системы и современной сети Web в систему 10.

Гибкое представление, которое передает семантику отображаемой информации.

Настоящее изобретение обеспечивает пользователей гибким представлением. Поскольку XML-Web-сервис посылает в ответ XML-данные, а не HTML-данные, и поскольку представление динамической генерируется у клиента, пользователь выбирает различные «поверхности» для просмотра семантической информации. Поверхности предпочтительно преобразуют XML в формат, пригодный для представления (например, XHTML+TIME, SVG и т.д.), позволяя пользователю динамически вбирать поверхности на ос-

нове возможностей различных технологий отображения. Например, формат SVG имеет множество свойств, которых нет у формата XHTML+TIME, и наоборот. Пользователь может выбрать SVG-Поверхность для сценариев, в которых формат SVG является оптимизированным. Альтернативно, пользователь также может выбрать формат XHTML+TIME для других сценариев.

Гибкость в выборе поверхностей как часть настоящего изобретения предоставляется приложениям в других ситуациях. В различных альтернативных вариантах осуществления используются поверхности типа «текст-речь», которые могут исполняться семантическим браузером 30 на второй (вспомогательной) машине одновременно с первой или основной машиной, например, для оказания помощи незрячим пользователям; поверхности с динамически изменяемыми размерами, которые адаптируются к размеру текущего порта представления (тем самым позволяя пользователю изменять размеры окна при сохранении приятного для пользователя опыта восприятия); поверхности, которые проверяют локальное состояние для отображения семантических подсказок (например, календарь пользователя в случае информации события, например, информация о свободном/занятом времени); поверхности, которые отображают внутристрочные окна предварительного представления, которые экономят пользователю время на навигацию и повышают производительность; поверхности, которые отображают различные настраиваемые подсказки для внутренних оповещений, срочных новостей, глубинной информации, интеллектуальных рекомендаций, внутренних связей, информации «лупы» и т.д. Пользователи также могут выбрать поверхности для использования с интеллектуальными хранителями экрана, например, когда пользователю желательно просматривать агента в режиме хранителя экрана. В альтернативном варианте система 10 поддерживает поверхности для шаблонов контекста (описаны выше), например, заголовков, ньюсмейкеров, обсуждений и т.д.

Ввиду возможности гибкого представления, настоящее изобретение позволяет пользователю выбрать наилучший режим представления на основе текущей задачи. Например, пользователи могут выбрать тонкую, «незаметную» поверхность при работе на их основной машине, когда производительность имеет более высокий приоритет, чем эстетический эффект. Пользователи могут выбрать «умеренную» поверхность в случаях, когда производительность важна, но и эффекты желательны или допустимы. Пользователи могут выбрать выразительную, привлекающую внимание поверхность для сценариев, подобных использованию вторичных машин, например, когда пользователи просматривают информацию периферийным зрением, и для них желательны свойства типа «текст-речь» для уведомления их о срочных новостях и т.д. Такие экспрессивные поверхности могут альтернативно отличаться анимацией, эффектами, подобными раскадровке, для глубинной информации, объектами, отображаемыми на траекториях движения, и другими специальными эффектами.

Кроме того, поверхности согласно настоящему изобретению факультативно конфигурируются для включения или исключения фильтров типа объекта. Например, поверхность может конфигурироваться для включения только «документов», но исключения «докладов аналитиков». Поскольку поверхность принимает XML-результаты для определения итогового представления, Поверхность может включать или исключать объекты в XML (SRML)-результатах на основе исследования типа объекта (или других признаков) возвращаемых объектов.

Логика, логический вывод, умозаключение.

Настоящее изобретение предусматривает логику, логический вывод и умозаключение. Модель семантических данных в агентстве KIS предпочтительно обеспечивает поддержку для логики посредством обработки базы данных семантической сети, преобразование семантических запросов в SQL и другие языки запросов баз данных для логической обработки и т.д. Кроме того, система 1-0 настоящего изобретения предпочтительно включает в себя машину логического вывода для логического вывода связей, например, экспертов по конкретной категории или информационному элементу, рекомендаций, вероятностных связей (например, вероятности, что некое лицо написало документ) и т.д. Как описано выше, машина логического вывода согласно настоящему изобретению предпочтительно наблюдает семантическую сеть, осуществляет в ней поиск для логического вывода новых семантических связей и представляет результирующие связи в таблице семантических связей.

Гибкий, управляемый пользователем анализ информации.

Настоящее изобретение предусматривает собственную поддержку для гибкого анализа информации у клиента. Презентатор в настоящем изобретении предпочтительно использует интеллектуальные лупы, чтобы обеспечить возможность пользователю предварительный просмотр результатов семантического запроса перед выдачей запроса. Пользователь может изменить релевантные предикаты и другие фильтры, чтобы предварительно просмотреть результаты. В альтернативном варианте осуществления пользователь имеет возможность активизировать запрос и использовать его как основу для нового подзапроса, если необходимо.

Гибкие семантические запросы.

Настоящее изобретение обеспечивает возможность пользователю выдавать весьма гибкие семантические запросы. Пользователь может вводить локальный контекст в запросы, например, путем использования фильтров, таких как «относится к этому документу на моем жестком диске». Ни современная сеть Web, ни семантическая сеть Web не обеспечивают этой возможности. Кроме того, настоящее изобре-

ние предпочтительно воплощает интеллектуальных агентов, которые используют ссылки на специализированном языке разметки семантических запросов (SQWML) и включают локальные и удаленные ресурсы, предикаты, ссылки на категории и объекты. Настоящее изобретение предпочтительно реализует простой для использования пользовательский интерфейс для создания и редактирования интеллектуальных агентов (представляющих семантические запросы), использующих простую модель мастера. Как описано выше, система 10 позволяет семантическим запросам формировать основу для новых запросов посредством рекурсивного применения функции «перетащить и оставить», например, документ или HTML-связь могут быть перемещены к существующим или новым интеллектуальным агентам, тем самым создавая последовательно новых интеллектуальных агентов. Интеллектуальные агенты альтернативно используются как «лупы», могут содержать объекты, «вставленные» в них для формирования новых семантических запросов, и могут быть добавлены к элементам сопряжения, которые сами по себе являются контейнерами семантических запросов и которые, в свою очередь, могут быть отфильтрованы путем создания при этом суб-элементов сопряжения или контейнеров субагентов.

Поддержка считывания/записи.

Система 10 настоящего изобретения предоставляет поддержку для функциональных средств считывания/записи за счет обеспечения XML-Web-сервиса, который позволяет пользователю публиковать информацию непосредственно в семантической сети.

Это может быть любым документом, аннотацией или семантической связью, которая корректирует прерванную связь или обеспечивает новую связь. Это также является субъектом ограничений защиты на XML-Web-сервисе и на уровне операционной системы. Система 10 использует аутентификацию, управление доступом и другие сервисы из операционной системы и сервера приложений, которые находятся под уровнем XML-Web-сервиса. Эти сервисы, обеспечивающие защищенность, предпочтительно используются для защиты доступа для считывания и записи в семантическую сеть.

Аннотации.

Настоящее изобретение предусматривает встроенную поддержку для аннотаций. Имеется специальный предикат «аннотировано (кем-либо)», который определяет семантическую связь аннотирования между объектом «персона» и любым другим информационным объектом (например, документом, почтовым отправлением, онлайн-курсом (обучения) и т.д.). Система 10 включает в себя поддержку уровня представления для аннотаций, обеспечивая возможность пользователю перемещаться к аннотациям через внутренние связи, «интеллектуальные лупы» и т.д. Способ, которым настоящее изобретение воплощает аннотации, обеспечивает преимущества существующих методов (таких как методы аннотирования «на месте», которые помещают аннотацию как часть информационного объекта, который аннотируется). В предпочтительном варианте настоящего изобретения аннотации являются «первоклассными» информационными объектами. Это означает, что они могут быть снабжены входящими и исходящими связями, «рассмотрены в лупу» (с использованием интеллектуальных луп), скопированы и вставлены (с использованием интеллектуального копирования и вставки) и т.д. Настоящее изобретение предоставляет аннотации для всех семантических инструментальных средств настоящего изобретения, тем самым оснащая пользовательский опыт более мощно, чем с использованием стандартных методов аннотирования. Кроме того, аннотации согласно настоящему изобретению используются с шаблонами контекста. В результате машина логического вывода может использовать их для увеличения интеллектуальности системы с течением времени. Кроме того, система 10 обеспечивает простое и легкое средство аннотирования объектов путем отправки специальным образом сформатированной электронной почты (с определенным телом сообщения) к агенту электронной почты агентства.

«Сеть Web доверия».

Настоящее изобретение обеспечивает «сеть Web доверия» посредством XML-Web-сервиса. Этот сервис аутентифицирует пользователя, который хочет обновить семантическую сеть, сделать утверждения, установить/обновить связи и т.д. Это также обеспечивает возможность предоставления обогащенного содержания через агентство KIS абонентам, зарегистрированным на получение платного содержания. Значимость всей сети возрастает, если пользователь может использовать те же самые инструментальные средства платформы для непрерывной навигации по множеству источников обогащенного содержания.

Информационные пакеты (элементы сопряжения).

Настоящее изобретение обеспечивает информационные пакеты или «элементы сопряжения». Элементы сопряжения представляют собой семантические контейнеры, которые включают в себя ссылки на семантические запросы от интеллектуальных агентов. Это позволяет пользователю обращаться со связанной семантической информацией как с целым блоком. Пользователь может отдельно просматривать индивидуальных агентов в элементах сопряжения или просматривать весь элемент сопряжения, как если бы информация в нем исходила от одного агрегированного агента. Это предпочтительно выполняется путем управления каждым агентом посредством обращения к XML-Web-сервису. В предпочтительном варианте осуществления пользователи перемещают с использованием операции «перетащить и оставить» объекты в элементы сопряжения для создания суб-элементов сопряжения. Это предпочтительно выполняется рекурсивно. Элементы сопряжения могут создаваться, удаляться и редактироваться. Пользователь может добавлять и удалять интеллектуальных агентов к/от элементов сопряжения. Элементы сопряже-

ния можно интерпретировать как цифровой эквивалент персональной газеты, которая содержит различные разделы. Например, такие печатные издания, как USA Today, New York Times, Wall Street Journal и т.д. содержат различные разделы, например новости, бизнес, спорт, жизнь/развлечения и т.д. Каждый из этих разделов соответствует записи интеллектуального агента в элементе сопряжения, и вся газета соответствует элементу сопряжения. Гибкий просмотр и перемещение, обеспечиваемые настоящим изобретением, могут интерпретироваться как цифровой эквивалент того, каким образом пользователь может просматривать каждый раздел газеты полностью и последовательно, по одному в каждый данный момент времени, или просматривать всю газету, начиная с первой страницы каждого раздела, переходя ко второй странице каждого раздела и т.д.

Шаблоны контекста.

Как описано более детально выше, настоящее изобретение обеспечивает шаблоны контекста, которые являются управляемыми сценарием шаблонами информационных запросов, которые отображаются на конкретные семантические модели для доступа к информации и ее извлечения. По существу, шаблоны контекста могут быть интерпретированы как персональные «каналы» извлечения цифровой семантической информации, которые доставляют информацию к пользователю путем использования предварительно определенного семантического шаблона. В предпочтительном варианте осуществления семантический браузер 30 позволяет пользователю создавать новый элемент сопряжения или специального агента с использованием шаблонов контекста для инициализации свойств агента. Шаблоны контекста предпочтительно агрегируют информацию на одном или более агентств. Кроме того, шаблоны контекста предпочтительно используются с палитрами контекста для обеспечения интеллектуального, динамического, непосредственного контекста для любого информационного объекта, который отображается или выбран пользователем.

Ориентированное на пользователя агрегирование информации.

Настоящее изобретение предусматривает собственную поддержку для ориентированного на пользователя агрегирования информации. Сценарии дают возможность пользователю просматривать контекст и зависимую от времени информацию, как если бы они происходили из одного источника, даже если они выделены из различных хранилищ информации. Это обеспечивает существенно более продуктивный пользовательский опыт по сравнению с современной сетью Web и концептуальной семантической сетью за счет обеспечения ориентированных на пользователя вычислений, причем пользователю представляется верная информация в правильном контексте и в соответствующее время, независимо от источника информации. Информационный агент агрегирует информацию динамическим образом, по разным информационным источникам, с использованием семантических запросов клиентской стороны посредством SQML и агрегированием XML-результатов, которые исходят из отклика на SQML разных агентств.

Е. Сценарии.

Ниже приведены примеры сценариев функционирования предпочтительных и альтернативных вариантов осуществления настоящего изобретения применительно к разным практическим ситуациям.

1. Примеры семантических запросов с использованием настоящего изобретения.

a. Найти весь контекст, относящийся к спецификации, которой соответствует путь к файлу `c:\spec.doc`

Необходимо перетащить и оставить пиктограмму, представляющую документ, на пиктограмму, представляющую информационного агента. Открывается файл в семантическом браузере и отображаются палитры контекста. В предпочтительном варианте они включают некоторые или все из следующих шаблонов контекста: заголовки, открытие, ньюсмейкеры, наступающие события, временная шкала, обсуждения, разное, образцы, наилучшие варианты выбора, сегодня, новости, вызывающие прерывание и т.д. Эти палитры включают в себя релевантный контекст из агентств в списках «последних» и «предпочтительных» в пространстве имен.

b. Найти всех экспертов в агентстве с названием "R&D", которые имеют опыт в беспроводной технологии.

Запустить мастера «новый интеллектуальный агент» и выбрать опцию «использовать шаблон контекста» при создании агента.

Выбрать агентство "R&D" из диалога «выбрать агентство» и выбрать категорию, называемую «беспроводная (технология)» из браузера категорий. Открыть вновь созданного интеллектуального агента.

c. Найти всю информацию Reuters, которая релевантна для связи на текущей просматриваемой Web-странице.

Перетащить и оставить связь с пиктограммой агентства, представляющей "Reuters". Создан новый интеллектуальный агент, названный «информация Reuters, релевантная для [заголовков связи]», и открытый в информационном агенте.

d. Найти всю информацию Reuters, которая релевантна для связи на текущей Web-странице и которая релевантна для спецификации, которой соответствует путь к файлу `c:\spec.doc`.

Перетащить и оставить пиктограмму, представляющую документ, к агенту, который только что создан выше «информация Reuters, релевантная для [заголовков связи]». Это создает нового интеллекту-

ального агента, озаглавленного «информация Reuters, релевантная для [заголовок связи] и релевантная для spec.doc». Это иллюстрирует управляемый пользователем просмотр ресурсов и динамическое связывание.

e. Найти всю электронную почту на внутреннем агентстве, озаглавленном «маркетинг», релевантную первой статье Reuters, которая была возвращена в предыдущем запросе.

Высветить объект «статья Reuters» и выполнить щелчок на кнопке для «Команд действия». Отображается всплывающее меню. Выбрать «копировать». Найти пиктограмму, представляющую агентство, озаглавленное «маркетинг» (на древовидном представлении расширения оболочки). Выполнить щелчок правой кнопкой на пиктограмме. Указать «вставить». Это создает и открывает нового интеллектуального агента под названием «информация по «маркетингу», релевантная [название статьи Reuters]». Активизировать блок в окне результатов, показывающий объекты электронной почты.

f. Перемещение к автору электронной почты.

Высветить объект электронной почты и выполнить щелчок на кнопке для «связей». Отображается всплывающее меню, показывающее внутренние (собственные) связи. Переместиться к элементу меню, озаглавленному «От:». Отображается всплывающее меню, показывающее объект «персона» на линии «от» объекта электронной почты. Выбрать желательный объект. Открывается новый интеллектуальный агент в информационном агенте, показывающий метаданные персоны, являющейся автором объекта электронной почты. Контекст персоны также отображается в палитрах контекста. Пользователи имеют возможность продолжать просмотр ресурсов с использованием объекта «персона» или его контекста (на любой из палитр контекста).

g. Перемещение к приложениям электронной почты.

Высветить объект электронной почты и выполнить щелчок на кнопке для «связей». Отображается всплывающее меню, показывающее внутренние связи объекта электронной почты. Переместиться к элементу меню, озаглавленному «приложения». Отображается всплывающее меню, показывающее названия приложений. Выбрать желательное приложение. Открывается приложение в качестве нового интеллектуального агента в окне информационного агента. Контекст для приложения отображается в палитрах контекста.

h. Найти все события в агентстве «события энергетической индустрии», которые релевантны приложению.

Высветить объект приложения и выполнить щелчок на кнопке для «команд действия». Отображается всплывающее меню. Выбрать «копировать». Найти пиктограмму, представляющую агентство, озаглавленное «события энергетической индустрии» (на древовидном представлении расширения оболочки). Выполнить щелчок правой кнопкой на пиктограмме. Указать «вставить». Это создает и открывает нового интеллектуального агента под названием «информация по «событиям энергетической индустрии», релевантная [название приложения электронной почты]».

i. Просмотр папки «Мои документы» с использованием Reuters в качестве контекста.

В информационном агенте выбрать «открыть документы в папке». Альтернативно, перетащить и оставить папку «Мои документы» на пиктограмму, представляющую информационного агента. Указать, должны ли быть включены подпапки. Это создает и открывает нового неинтеллектуального агента под названием «Мои документы». Если выполнить щелчок на этом агенте, то метаданные для документов в этой папке открываются в информационном агенте. Если один из этих документов выбран, то отображаются палитры контекста для этого документа. Для просмотра документов с использованием Reuters в качестве контекста пользователь находит пиктограмму, представляющую агентство Reuters, выполняет щелчок правой кнопкой на пиктограмме и указывает «копировать». Пользователь указывает на любой из результатов, показывающих метаданные документа в информационном агенте и выбирает пиктограмму, указывающую интеллектуальную лупу. Окно интеллектуальной лупы отображается, показывая информацию в результатах реляционного запроса. Количество элементов, найденных в Reuters, которые релевантны документу, отображается в дополнение к информации, такой как самый последний посланный элемент. Кроме того, отображается средство управления предварительным просмотром, позволяя пользователю предварительно просмотреть результаты по ходу. Пользователь может выбрать выполнение щелчка на результатах, чтобы открыть агента, представляющего новый реляционный запрос. В ответ отображается контекст для первого объекта в результатах с использованием палитры контекста.

j. Уведомление посредством электронной почты, речевой почты или на пейджер, если имеются новости, вызывающие прерывание, которые относятся к предмету в технологии XML и которые относятся к данному документу.

Создать нового интеллектуального агента с использованием категории «XML» в качестве фильтра категории. Перетащить и оставить пиктограмму, представляющую этот документ, к агенту. Это создает нового интеллектуального агента с соответствующим названием. Перейти к меню «опции» в информационном агенте и ввести надлежащую информацию в раздел уведомления (ваш адрес электронной почты, номер пейджера, телефонный номер и т.д.). Выполнить щелчок правой кнопкой на интеллектуальном агенте и выбрать «уведомить».

2. Проблемы бизнеса.

а. Доступ к информации.

Современная сеть Web. Джон Хед-Мастер работает в FastServe, компании маркетинговых консалтинговых услуг в Сан-Диего. Каждый день он приходит на работу и запускает свой Web-браузер. В этот день он решает просмотреть ресурсы корпоративной сети Web, чтобы проверить, имеется ли новая, представляющая интерес информация. Исходная страница браузера установлена (с использованием информационного портала предприятия) на корпоративную исходную страницу. Корпоративная исходная страница имеет связи с исходными страницами различных отделений компании. Джон перемещается к этим связям и от них выполняет выбор связей щелчком кнопки мыши. Спустя некоторое время он приходит в полное разочарование, поскольку он знает, что имеется больше источников информации, к которым он не может переместиться лишь потому, что он не знает к ним путей. В конечном счете он сдаётся.

Информационная нервная система. Джон запускает своего информационного агента (семантический браузер). Это открывает исходного агента. На странице он видит список связей знаний, соответствующих продуктам, группам продуктов, докладам, корпоративным событиям, онлайн-курсам обучения, видеопрезентациям. Он указывает на связь «группы продуктов». Автоматически появляется всплывающий контур, показывающий количество групп продуктов и другие данные, относящиеся к связи. Затем он открывает связь. Список объектов группы продуктов отображается с настраиваемым представлением или «поверхностью». Затем он наводит свою мышь на первый из них. Немедленно над связью появляется всплывающее меню с действиями «Показать членов группы», «Перечислить сходные группы продуктов», «Подписаться на события группы». Он выполняет щелчок на опции «Подписаться на события группы» и теперь будет уведомляться по электронной почте (через информационного агента предприятия) обо всех событиях, которые относятся к этой группе продуктов. Затем он выполняет щелчок на опции «Показать членов группы». Это открывает новую «Страницу знаний» с пиктограммами, соответствующими людям. Он указывает на пиктограмму для Сьюзен Групп-Лидер. Всплывающий контур представляет информацию о Сьюзен. Появляется меню правой кнопки с действиями, опциями «сообщает (кому-либо)», «список прямых сообщений», «член (чего-либо)», «созданные документы», «недавно посещенные собрания». Джон затем выбирает «недавно посещенные собрания». Это открывает новую страницу знаний с одним объектом «собрание». Джон указывает ее и продолжает просмотр.

В некоторый момент Джон решает отыскать сотрудника, с которым он встречался накануне. Он набирает «Уилбер Джонс». Это возвращает объект «персона», соответствующий Уилберу. Джон продолжает просмотр с использованием Уилбера в качестве опорного пункта информационных знаний.

В конечном счете Джон осознает, что Уилбер, по-видимому, не имеет информации, требуемой Джону. Джон затем вводит следующий запрос в окне поиска в своем информационном агенте: «Перечислить все онлайн-курсы обучения и документы, которые относятся к собраниям, посвященным продажам в 2002». Информационный агент (посредством агента электронной почты) возвращает список онлайн-курсов обучения и документов, которые соответствуют запросу знания.

б. Управляемое знаниями управление связями с клиентами.

«Точки соприкосновения» (контакты) с клиентами. AnySoft является производителем программного обеспечения с 50 продуктами на 100 различных языках. Они используют свой Web-сайт для обеспечения обновленной информации для своих клиентов. Однако клиенты жалуются, что их Web-сайт очень труден для навигации, и считают, что очень трудно найти информацию о продуктах и подписаться на уведомления.

Путем развертывания информационной нервной системы согласно возможному варианту осуществления изобретения компания AnySoft развернула информационную нервную систему, которая сосуществует с имеющимся Web-сайтом. Информационный агент доступен с исходной страницы и из строки поиска. Клиенты теперь имеют намного более интуитивный способ навигации по Web-сайту для поиска продуктов, «белых книг» (официальных изданий), уведомлений, пресс-релизов, корпоративных событий и т.д. Клиенты теперь могут вводить запросы на естественном языке, которые возвращают объекты знания, отличающиеся самоуправляемостью и действенностью. Уже только эти признаки дают клиентам доступ к знаниям «на их кончиках пальцев». Клиенты также могут использовать естественный язык для перемещения по сайту компании AnySoft со своих портативных устройств.

Обратная связь и отслеживание клиентов. Comp-Map является торговым посредником по поставкам компьютерных периферийных устройств с множеством каналов распространения продукции. Компания получает обратную связь от клиентов со своего Web-сайта, своего центра обработки заказов, непосредственно своих продавцов, своих агентов телемаркетинга и т.д. Эта обратная связь поступает в форме документов и электронной почты. Компания выявила проблему, состоящую в том, что клиентская обратная связь не направляется надлежащим образом в компании к тем людям, которым нужна эта информация. Служащие в отделении по разработке продукции жалуются руководству, что для них трудно интегрировать клиентскую обратную связь в процесс разработки продукции, поскольку они не знают, где найти соответствующую информацию и из-за того, что важные знания не являются совместно используемыми в организации.

При развертывании информационной нервной системы электронная почта, которая содержит кли-

ентскую обратную связь, теперь поступает семантически интегрированной в семантическую среду компании. KIS настоящего изобретения автоматически добавляет семантические связи между электронной почтой клиентской обратной связи и семантическими объектами, такими как документы, проекты и служащие, которые работают над соответствующими продуктами. Клиентская обратная связь интеллектуальным образом «всплывает» в нужных местах в пространстве знаний. Агент электронной почты посылает периодические уведомления людям, которые, по всей вероятности, должны быть заинтересованы в прочтении электронной почты клиентской обратной связи.

Также с использованием информационной нервной системы клиент становится опорным центром информационного знания. Это позволяет намного быстрее и проще воздействовать на клиентскую обратную связь и отслеживать связанные с клиентами знания в рамках всей организации. Информационная нервная система автоматически аннотирует объект «Клиент» с релевантными сообщениями электронной почты, документами, сходными клиентами и т.д. За счет этого связи с клиентами могут пересылаться по электронной почте, и сослуживцы могут начать поиск релевантной информации с этих связей. Объект «Клиент» обеспечивает возможность поиска, просмотра ресурсов и т.д.

c. Управляемые знаниями прямые связи с розничной торговлей и обслуживание на месте продажи.

Марша Миндсет является агентом по обслуживанию клиентов JustIn Time Support Service, фирмы компьютерного сервиса в Канзас-Сити, шт. Миссури. Марша посещает клиентов в регионе Канзас-Сити и всегда берет с собой свой беспроводный портативный компьютер (PDA-персональный цифровой помощник), так что она может посылать электронную почту во вспомогательные головные офисы всякий раз, когда она испытывает затруднения. Компания JustIn Time недавно развернула KIS и агента электронной почты. Теперь, всякий раз для решения необходимых вопросов, Марша может послать электронную почту агенту электронной почты и задать вопросы на естественном языке. Агент электронной почты отвечает на ее электронную почту прямым ответом или «связями знаний», которые позволяют Марше получить мгновенный доступ к релевантным объектам электронной почты, документам или людям, которым она затем может послать электронную почту или позвонить по телефону. Сотрудники отделения прямых связей с розничной торговлей JustIn Time также используют технологию настоящего изобретения в решении вопросов при обслуживании клиентов на месте продажи. Представители службы прямых связей с розничной торговлей также имеют при себе беспроводные PDA и могут вводить запросы к агенту электронной почты.

d. Анализ отдельных проблем.

Корпоративное обучение, перенос и совместное использование знаний. WaveGen является компанией, специализирующейся в области биотехнологии, обеспечивающей «управляемое разрешение проблем» для медицинских работников в США. Компания недавно развернула платформу системы управления обучением Saba для подготовки своих сотрудников (особенно представителей службы продаж). Это способствует экономии дорожных расходов и позволяет продавцам компании быть лучше подготовленными к обслуживанию медицинских работников в различных регионах страны. Она также позволяет исследователям компании всегда быть информированными о последних открытиях в отрасли исследований по биотехнологии.

Компания также имеет другое программное обеспечение, позволяющее поддерживать ценные источники знаний. Она развернула средства управления информационным содержанием, которые включают в себя документы, медийные файлы, Microsoft Exchange для электронной почты и программное обеспечение для совместного использования для онлайн-конференций. Однако компания установила, что перенос знаний в ней не очень эффективен, поскольку он не интегрирует все эти имеющиеся средства. Представители службы продаж отметили, что они не имеют средств для обнаружения важных источников знаний внутри и вне организации для оказания им помощи в выставлении на продажу продуктов компании для ознакомления с ними медицинских работников. В настоящее время используются информационные порталы предприятия для информирования продавцов о предстоящих онлайн-курсах для обучения и о важных событиях. Однако представители службы продаж жалуются на то, что большая часть знаний (сохраненных в электронной почте, документах и т.д.) не доводится до их сведения, поскольку никто не знает, кому еще они могут быть полезны.

Кроме того, представители службы продаж используют Microsoft Outlook для добавления в свои календари назначений, относящихся к предстоящим визитам специалистов. Однако они жалуются на то, что они только получают напоминания об этих назначениях, и что масса информации, которая могла бы им помочь в более эффективной продаже продукции, не предоставляется им автоматически перед соответствующими назначенными визитами специалистов.

Компания WaveGen недавно развернула информационного агента на основе технологии согласно настоящему изобретению. Компания развернула KIS и агента электронной почты для облегчения интеллектуальных информационных соединений и маршрутизации информации для оказания помощи своим сотрудникам из подразделений продаж и исследований в целях принятия ими более эффективных решений при обслуживании клиентов и совершенствовании продукции компании. С использованием информационного агента продавцы имеют мгновенный доступ не только к документам, но и к «объектам знаний», которые более непосредственно связаны с решаемыми ими конкретными задачами. Например,

представители службы продаж теперь имеют агента «Доктор Джонс» в качестве XML-объекта. Это не документ и не Web-страница. Напротив, это семантическое представление клиента. Представители службы продаж могут просматривать семантические связи, такие как «последние сообщения электронной почты», «релевантные документы», «свойства», «важные данные», «релевантные предстоящие онлайн-курсы обучения» и т.д. Таким путем клиент становится опорным пунктом, с использованием которого агент службы продаж перемещается во внутренней сети Web. Эти связи могут генерировать результаты из совместного использования файлов, сохраненной электронной почты, Microsoft Exchange и т.д. Но вместо поиска или перемещения к этим источникам знаний как к отдельным островкам, представитель службы продаж может обнаружить новое знание на основе семантических взаимосвязей, относящихся к решаемой им конкретной задаче.

Таким способом представитель службы продаж может иметь намного более эффективные знания «на кончиках пальцев», что способствует лучшему обслуживанию клиентов. Это знание исходит от сотрудников, из документов, которые были опубликованы другими агентами службы продаж, электронной почты, посланной по спискам распределения, о существовании которых может быть неизвестно, и т.д. KIS осуществляет интеллектуальную функцию путем автоматического установления семантических соединений из всех этих несопоставимых источников. Это становится затем весьма эффективной формой совместного использования знаний, поскольку разные сотрудники могут перемещаться в информационном агенте с использованием одного и того же опорного пункта «Доктор Джонс».

Агент электронной почты также позволяет представителям службы продаж выдавать запросы о знаниях на естественном языке. Результаты запросов выводятся из машины логического вывода и могут основываться на знании, которое было логически выведено из существующих знаний.

Высокоэффективным средством информационной нервной системы настоящего изобретения является перенос, совместное использование знаний и открытие всего, что происходит, автоматически на основе семантической сети.

3. Ситуации.

а. Открытие, извлечение, доставка и просмотр семантической информации.

Джо Нолидж-Уоркер запускает информационного агента (основанный на XML семантический браузер настоящего изобретения).

Когда он входит в систему, ему представляется подсказка посредством диалогового окна, что в семантическом интранете имеются новые агенты. Затем он просматривает список агентов из собственной организации и извне, который может включать следующее:

- Документы.Технология.Все
- Документ.Маркетинг.Все
- Люди.Подразделения.Продажи.Все
- Люди.Подразделения.Продажи.Менеджеры
- ОнлайнныеКурсы.Продажи.101
- ОнлайнныеКурсы.Технология.XML.101
- Собрания.ЭтаНеделя.Все
- Собрания.ПрошлаяНеделя.Все
- Книги.Компьютеры.Программирование.Все
- НовостныеГруппы.Microsoft.Общедоступные.SOAP
- ЭлектроннаяПочта.Источники.Все
- ЭлектроннаяПочта.Источники.ПроектX.Все
- События.Технология.Беспроводная.Все
- Отчеты.Gartner.ПрограммноеОбеспечение.Все
- Отчеты.IDC.Все
- Видео.ExecutivePresentations.Все

Затем он выбирает «Собрания.ЭтаНеделя.Все» информационный агент отображает список объектов, который представляет собрания, которые он посетил на этой неделе. Эта информация поступает из Microsoft Exchange, но это не было представлено ему. Затем Джо указывает на связь для первого объекта «собрание». Отображается всплывающий контур, в котором сообщается, что новый курс обучения только что объявлен для доступа в интранете. Также всплывающий контур указывает, что на IDC имеется новый отчет, который может представлять интерес для Джо. В дополнение к всплывающему контуру, справа от объекта отображается всплывающее меню. Меню имеет следующие команды действия:

- Перечислить участников
- Перечислить возможные замены участников
- Показать связанные объекты →
- Новости.Reuters.ПрогнозРынка.Все
- Документы.Технология.Все
- События.Корпоративные.Сегодня.Все
- Подписаться на рассылку для напоминания

Затем Джо выбирает опцию «Подписаться на рассылку для напоминания». Это устанавливает кон-

такт с агентом рассылки напоминаний о собраниях на сервере. Этот агент затем посылает периодические обновления релевантной информации участникам собрания. Это может быть сделано как посредством браузера или посредством электронной почты. Джо затем выбирает связанные объекты в опции «События.Корпоративные.Сегодня.Все».

Отображается список объектов, информирующих о событиях. Джо указывает на первый объект, и отображается всплывающее меню.

Джо выбирает «Добавить в календарь», и событие добавляется в его календарь. Затем Джо решает, что ему нужно найти все события в отрасли, которые связаны с корпоративным событием. Он перемещает этот объект к агенту «События.Технология.Все» с помощью мыши и оставляет его. После оставления объекта, браузер загружает информационные объекты из агента «События.Технология.Все» (по разным Web-сайтам и другим островкам информации), которые связаны с корпоративным событием, объект которого он перетащил.

На следующей неделе Джо получает электронную почту от агента электронной почты. В электронной почте агент информирует Джо, что им отмечено, что каждый, кто добавил событие к своему календарю, также наблюдал видеоматериалы корпоративного обучения с корпоративного медиа-сервера. Сообщение электронной почты содержит XML-связь, которая приводит Джо назад к информационному агенту. Браузер затем отображает метаданные для видео. Одним из элементов всплывающего меню является «Просмотреть видео». Джо затем выбирает его и просматривает видео.

Следующий раз, когда Джо входит в систему на своей рабочей станции, он замечает, что имеются новые агенты. Затем он подписывается на опцию «Книги.Ebay.Компьютеры.Все» и добавляет это в свой список «Мои агенты». Автоматически в одном варианте осуществления изобретения этот агент добавляется в семантическую среду для Джо. Информационный агент выполняет неявные запросы и обеспечивает рекомендации (ранжированные по релевантности и временной зависимости), которые включают этого агента. Затем он выполняет щелчок на этом агенте и в панели результатов отображаются семантические информационные объекты (представляющие книги). Когда он указывает на один из объектов, немедленно отображается всплывающий контур, уведомляющий его, что намечена отраслевая конференция, проводимая автором книги. Если он выполняет щелчок по всплывающей связи, то объект события загружается в браузер и завершается командами действия, которые позволяют ему добавить событие в свой календарь (либо Microsoft Outlook, либо календарь на основе Интернет, подобный MSN календарю (доступный через сервисы Hailstorm Web компании Microsoft), AOL календарь и т.д.).

Пояснение сценария.

Этот сценарий показывает, каким образом с помощью настоящего изобретения специалисты в области информационных технологий могут получить доступ к «интегрированным знаниям». В данном примере компания, где работает Джо, «импортировала» агентов знаний от Gartner, IDC, Reuters, Ebay и т.д. в свое пространство знаний. Эти агенты автоматически добавляют знания в семантическую сеть компании. Сценарий также показал, каким образом Джо имел возможность получить представление «модели объекта» всего пространства знаний организации через интуитивно именованных интеллектуальных объектов. Джо имел возможность использовать этих агентов для «входа» в семантическую среду и затем навигации в ней своим путем из этой точки. Все информационные объекты доставлялись в реальном времени и были действенными (с соответствующими командами действий, которые отображались по ходу). Таким путем Джо не нужно было заботиться о том, из каких островков информации были получены объекты или какие приложения генерировали их.

Сценарий также показывает, каким образом Джо имел возможность обнаруживать не только новую информацию, но и новых агентов. Сценарий также показывает объединение знаний в действии - посредством совместной фильтрации - когда информационный агент давал рекомендации Джо на основе того, что он еще отмечал в работе предприятия.

Наконец, сценарий иллюстрирует, каким образом зависимость от времени информация автоматически доводится до сведения пользователя в том месте контекста, где это имеет смысл. Агент электронной почты автоматически связывал книгу, полученную из Ebay с предстоящим отраслевым событием, логически выводил и присваивал релевантность и ранг временной зависимости для события и принимал решение, являлось ли событие достаточно критическим, чтобы обеспечить немедленное отображение информации посредством оповещения семантического браузера.

b. Одноранговое совместное использование и сбор знаний.

Нэнси Хард-Уокер работает в компании Fortune 500, насчитывающей 40000 служащих. Она подписана на множество сайтов и использует информацию, пересылаемую ей по электронной почте друзьями и сослуживцами. Она только что получила некоторый объем информации и хотела бы поделиться ею с организацией. Она посылает документы во все списки распределения, членом которых является. Агент информации предприятия также является членом этих списков (агент добавляет себя ко всем общедоступным спискам распространения при инсталляции сервера). Когда агент получает информацию, он классифицирует ее и добавляет ее в семантическую сеть. Машина логического вывода затем подбирает информацию.

Несколько тысяч сотрудников не являются членами списков распределения, в которые Нэнси по-

слала документы. Однако они все используют интегратора и все подписались на агента «Электронная почта.Общедоступная.Все». Когда они просматривают другие связанные части сеть Web знаний, отображается всплывающий контур, указывающий, что имеется новая релевантная электронная почта у агента «Электронная почта.Общедоступная.Все». Сотрудники затем открывают этого агента, и объект электронной почты отображается. Один из элементов меню электронной почты соответствует следующему: «показать список распределения, в который было направлено сообщение». Сотрудники выбирают этот элемент, и информационные объекты списка распределения отображаются в браузере. Сотрудник затем указывает на список распределения, и отображается элемент всплывающего меню. Первый элемент соответствует «показать членов списка», второй - «присоединить». Сотрудники затем осуществляют присоединение к списку распределения.

Пояснение сценария. Этот сценарий иллюстрирует, каким образом информация была опубликована, совместно использовалась и собиралась посредством электронной почты и каким образом за счет использования семантической сети другие сотрудники оказались осведомленными об этой информации (и о списке распределения, о существовании которого он ничего не знали) под разными, но связанными «аспектами знаний». Сценарий показывает одноранговое совместное использование знаний полностью непрерывным, интегрированным способом, который не требует от пользователей публиковать информацию в хранилищах или самостоятельно классифицировать информацию. В некоторых вариантах осуществления изобретения все осуществляется автоматически (на заднем плане) и знания «всплывают» в нужных местах.

Хотя предпочтительный вариант осуществления и альтернативные варианты осуществления изобретения проиллюстрированы и описаны, как изложено выше, различные изменения могут быть осуществлены без отклонения от сущности и объема изобретения. Соответственно, объем настоящего изобретения не ограничивается раскрытием предпочтительного или альтернативных вариантов осуществления.

ФОРМУЛА ИЗОБРЕТЕНИЯ

1. Система для извлечения, управления, доставки и представления знаний, содержащая первый сервер, программируемый для добавления и поддержания семантической информации, специфической для предметной области;

второй сервер, осуществляющий связь с первым сервером, причем второй сервер программируется для содержания информации, специфической для предметной области, используемой для классификации и категоризации семантической информации;

клиент, обеспечивающий пользовательский интерфейс для пользователя для осуществления связи с первым и вторым серверами,

при этом первый и второй серверы работают совместно для выполнения этапов

обеспечения защиты информации из источников информации;

семантического связывания информации из источников информации;

поддержания семантических атрибутов семантически связанной информации;

доставки запрошенной семантической информации на основе пользовательских запросов и

представления семантической информации в соответствии с настраиваемыми пользовательскими предпочтениями.

2. Способ извлечения, управления, доставки и представления знаний для использования с серверной системой, программируемой для добавления, поддержания и сохранения семантической информации, специфической для предметной области, которая используется для классификации и категоризации семантической информации, включающий в себя

обеспечение защиты информации из источников информации;

семантическое связывание информации из источников информации;

поддержание семантических атрибутов семантически связанной информации;

доставку запрошенной семантической информации на основе пользовательских запросов и представление семантической информации в соответствии с настраиваемыми пользовательскими предпочтениями.

3. Система управления информацией, содержащая

память, имеющую возможность хранения объектов, имеющих схему и семантические связи;

процессор, осуществляющий связь с памятью, причем процессор выполнен с возможностью

добавления объектов к памяти;

удаления объектов из памяти;

создания сети, состоящей из объектов, имеющих семантические атрибуты и семантические связи между объектами на основе набора предварительно определенных правил для определения взаимоотношений между объектами;

поддержания семантических атрибутов объектов и семантических связей между объектами сети в памяти.

4. Система управления информацией, содержащая

сервер, конфигурируемый для управления семантической информацией, используемой для классификации и категоризации семантической информации;

клиент, обеспечивающий пользовательский интерфейс для пользователя для осуществления связи с сервером,

при этом система работает для выполнения этапов

обеспечения защиты информации из источников информации;

семантического связывания информации из источников информации с использованием семантической информации, имеющейся на сервере;

поддержания семантических атрибутов семантически связанной информации;

доставки семантической информации на основе пользовательских запросов и

представления семантической информации в соответствии с настраиваемыми пользовательскими предпочтениями.

5. Система управления информацией, содержащая

память, имеющую возможность хранения по меньшей мере одного объекта, имеющего схему и семантические связи и содержащего специфическую для предметной области семантическую информацию, используемую для классификации и категоризации информации;

процессор, осуществляющий связь с памятью, причем процессор выполнен с возможностью

добавления объекта к памяти;

удаления объекта из памяти;

создания сети, имеющей семантические атрибуты и семантические связи, ассоциированные по меньшей мере с одним объектом на основе специфической для предметной области информации в памяти;

поддержания семантических атрибутов объектов и семантических связей по меньшей мере с одним объектом сети в памяти.

6. Система управления информацией, содержащая

сервер, конфигурируемый для добавления, поддержания и хранения специфической для предметной области семантической информации, используемой для классификации и категоризации информации;

клиент, обеспечивающий пользовательский интерфейс для пользователя для осуществления связи с сервером,

при этом система работает для выполнения этапов обеспечения защиты информации из источника информации;

семантического связывания информации из источников информации с использованием специфической для предметной области семантической информации, имеющейся на сервере;

поддержания семантических атрибутов семантически связанной информации;

доставки семантической информации на основе пользовательского запроса и

представления семантической информации в соответствии с настраиваемыми пользовательскими предпочтениями.

7. Система представления информации, содержащая

сервер, хранящий семантическую информацию, используемую для классификации и категоризации информации;

клиент, обеспечивающий пользовательский интерфейс для пользователя для осуществления связи с сервером;

при этом система работает для выполнения этапов

доставки семантической информации на основе пользовательского запроса и

представления семантической информации в соответствии с настраиваемыми пользовательскими предпочтениями.

8. Система для извлечения, управления, доставки и представления знаний, содержащая

первый сервер, выполненный с возможностью управления семантической информацией;

второй сервер, осуществляющий связь с первым сервером, причем второй сервер выполнен с возможностью содержания информации, специфической для предметной области, используемой для классификации и категоризации семантической информации;

клиент, обеспечивающий пользовательский интерфейс для пользователя для осуществления связи с первым и вторым серверами,

при этом система работает совместно для выполнения этапов обеспечения защиты информации из источников информации;

семантического связывания информации из источника информации с использованием информации, специфической для предметной области, имеющейся на втором сервере;

поддержания семантических атрибутов семантически связанной информации;

доставки семантической информации на основе пользовательского запроса и

представления семантической информации в соответствии с настраиваемыми пользовательскими предпочтениями.

9. Способ динамического формулирования запроса семантической информации относительно по меньшей мере одного объекта, имеющего схему, на основе запроса на получение информации от пользователя, имеющего устройство ввода, взаимосвязанное с пользовательским интерфейсом, включающий в себя

прием от пользователя запроса на получение информации относительно по меньшей мере одного объекта;

создание запроса на основе упомянутого запроса на получение информации;

извлечение метаданных из объекта на основе схемы объекта;

видоизменение запроса на основе извлеченных метаданных и

активизацию видоизмененного запроса.

10. Система управления информацией, содержащая

сетевой компонент, имеющий память, конфигурированную для управления объектами, имеющими схему и семантические связи;

компонент сбора семантических данных, осуществляющий связь с сетевым компонентом, причем компонент сбора семантических данных конфигурирован для управления семантической информацией относительно объектов в памяти сетевого компонента и

компонент процессора семантических запросов, осуществляющий связь с сетевым компонентом и клиентом.

11. Система для обеспечения семантической информации клиенту, содержащая

источник информации;

память, содержащую объекты и семантические связи;

компонент процессора, осуществляющий связь с памятью, причем компонент процессора программируется для семантического связывания информации из источника информации с использованием памяти объектов и семантических связей и

компонент агента, который выдает семантически связанную информацию из источника информации в ответ на запрос клиентом.

12. Система для обеспечения семантической информации в ответ на пользовательский запрос, содержащая

множество компонентов памяти, содержащих объекты и семантические связи;

компонент агента, ассоциированный с предварительно определенным запросом на получение информации конкретного типа, причем компонент агента конфигурирован для создания по меньшей мере одного подзапроса на основе упомянутого пользовательского запроса;

извлечения информации из множества компонентов памяти, ассоциированной с подзапросом;

упорядочения информации, извлеченной из множества компонентов памяти, и

обеспечения пользователя семантически связанной информацией в ответ на упомянутый пользовательский запрос на основе упорядоченной информации, извлеченной из множества компонентов памяти.

13. Способ извлечения, управления, доставки и представления знаний для использования с серверной системой, программируемой для добавления, поддержания и сохранения информации, которая используется для классификации и категоризации семантической информации, и базой данных типов объектов и семантических связей, включающий в себя

прием запроса от клиента относительно семантической информации, ассоциированной с выбранным объектом;

если семантическая информация, ассоциированная с выбранным объектом, доступна в базе данных, то использование серверной системы для определения семантической информации, ассоциированной с выбранным объектом, на основе базы данных типов объектов и семантических связей, и

представление семантической информации, ассоциированной с выбранным объектом.

14. Система для извлечения, управления, доставки и представления знаний, содержащая

сервер, программируемый для добавления, поддержания и хранения семантической информации, специфической для предметной области, используемой для классификации и категоризации семантической информации, при этом сервер работает для выполнения этапов

обеспечения защиты информации из источников информации;

семантического связывания информации из источников информации;

аннотирования информации со специфической для предметной области информации;

поддержания семантических атрибутов семантически связанной информации и

доставки запрошенной семантической информации на основе пользовательских запросов и

клиент, обеспечивающий пользовательский интерфейс для пользователя для осуществления связи с сервером, при этом клиент работает для выполнения этапов

приема пользовательского запроса;

получения семантической информации от сервера на основе метаданных, извлеченных из пользовательского запроса; и

представления семантической информации в соответствии с настраиваемыми пользовательскими предпочтениями.

15. Система управления информацией, содержащая первый сервер, конфигурируемый для управления семантической информацией; второй сервер, осуществляющий связь с первым сервером, при этом второй сервер конфигурирован для хранения специфической для предметной области информации, используемой для классификации и категоризации семантической информации;

клиент, обеспечивающий пользовательский интерфейс для пользователя для осуществления связи по меньшей мере с одним из первого и второго серверов; при этом система работает для выполнения этапов обеспечения защиты информации из источника информации; семантического связывания информации из источника информации с использованием специфической для предметной области семантической информации, имеющейся на втором сервере; поддержания семантических атрибутов семантически связанной информации; доставки семантической информации на основе пользовательского запроса и представления семантической информации в соответствии с настраиваемыми пользовательскими предпочтениями.

16. Система для управления информацией, содержащая множество первых серверов, конфигурированных для управления семантической информацией; второй сервер, осуществляющий связь по меньшей мере с одним из первых серверов, причем второй сервер конфигурирован для содержания информации, специфической для предметной области, используемой для классификации и категоризации семантической информации;

клиент, обеспечивающий пользовательский интерфейс для пользователя для осуществления связи по меньшей мере с одним из первых серверов и вторым сервером, при этом система работает для выполнения этапов обеспечения защиты информации из источника информации; семантического связывания информации из источника информации с использованием специфической для предметной области информации, имеющейся на втором сервере; поддержания семантических атрибутов семантически связанной информации; доставки семантической информации на основе пользовательского запроса и представления семантической информации в соответствии с настраиваемыми пользовательскими предпочтениями.

17. Система для управления информацией, содержащая первый сервер, конфигурированный для управления семантической информацией; множество вторых серверов, осуществляющих связь с первым сервером, причем по меньшей мере один из вторых серверов конфигурирован для содержания информации, специфической для предметной области, используемой для классификации и категоризации семантической информации;

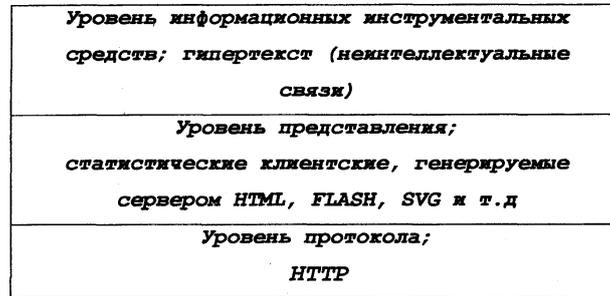
клиент, обеспечивающий пользовательский интерфейс для пользователя для осуществления связи по меньшей мере с одним из первого и вторых серверов, при этом система работает для выполнения этапов обеспечения защиты информации из источника информации; семантического связывания информации из источника информации с использованием специфической для предметной области информации, имеющейся по меньшей мере на одном из вторых серверов; поддержания семантических атрибутов семантически связанной информации; доставки семантической информации на основе пользовательского запроса и представления семантической информации в соответствии с настраиваемыми пользовательскими предпочтениями.

18. Система для управления информацией, содержащая множество первых серверов, конфигурированных для управления семантической информацией; множество вторых серверов, осуществляющих связь по меньшей мере с одним из первых серверов, причем по меньшей мере один из вторых серверов конфигурирован для содержания информации, специфической для предметной области, используемой для классификации и категоризации семантической информации;

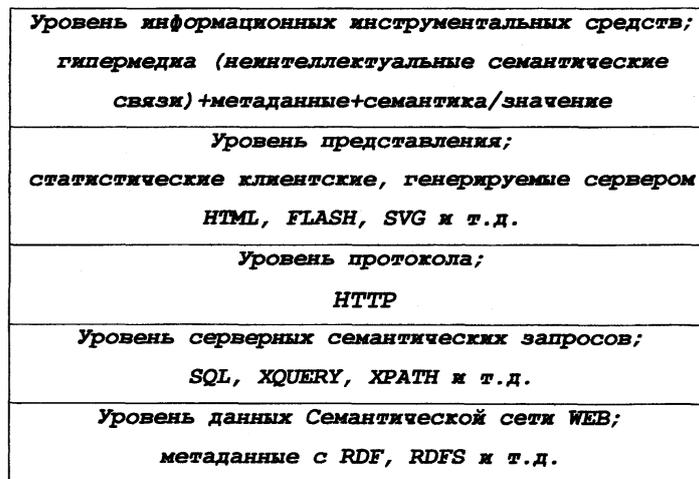
клиент, обеспечивающий пользовательский интерфейс для пользователя для осуществления связи по меньшей мере с одним из первых и вторых серверов, при этом система работает для выполнения этапов обеспечения защиты информации из источника информации; семантического связывания информации из источника информации с использованием специфической для предметной области информации, имеющейся по меньшей мере на одном из вторых серверов; поддержания семантических атрибутов семантически связанной информации; доставки семантической информации на основе пользовательского запроса и представления семантической информации в соответствии с настраиваемыми пользовательскими предпочтениями.

19. Система управления информацией, содержащая

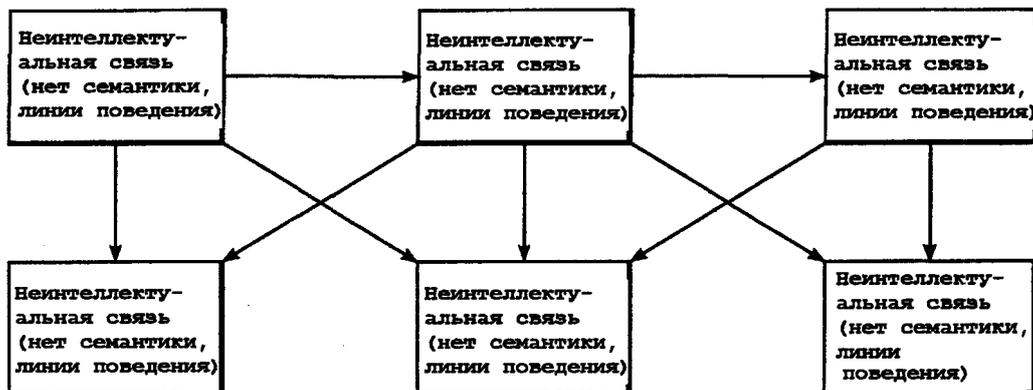
средство для сохранения объектов и семантических связей;
 средство для семантического связывания объектов и информации;
 средство для поддержания семантической информации, относящейся к объектам и семантическим связям; и
 средство для передачи семантической информации клиенту.



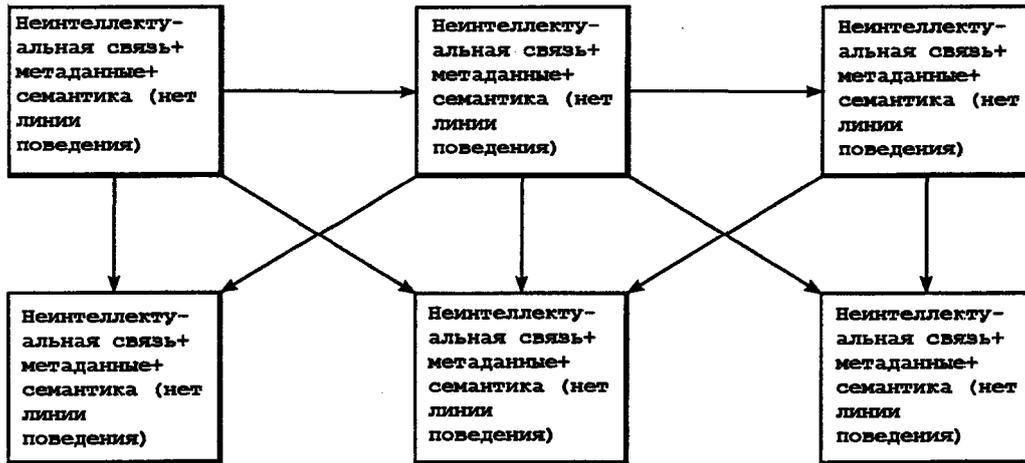
Фиг. 1



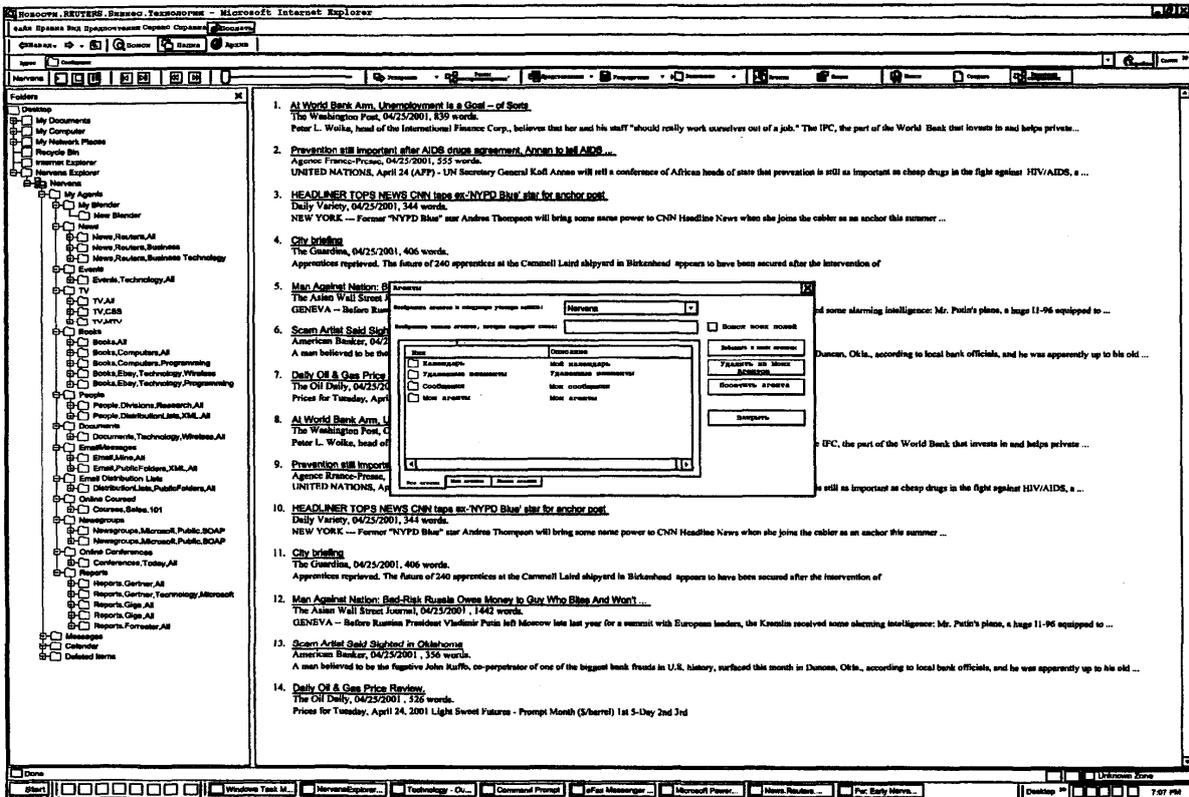
Фиг. 2



Фиг. 3



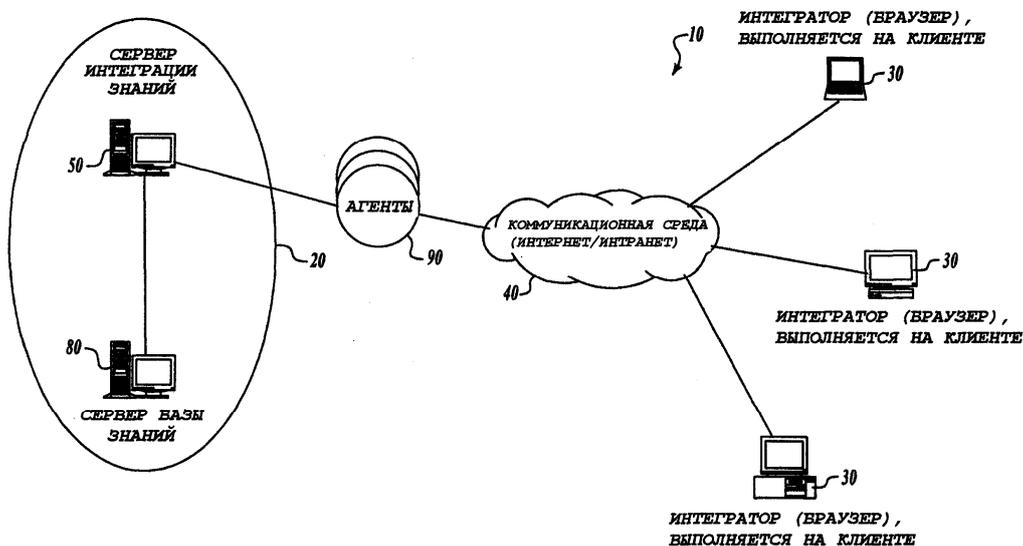
Фиг. 4



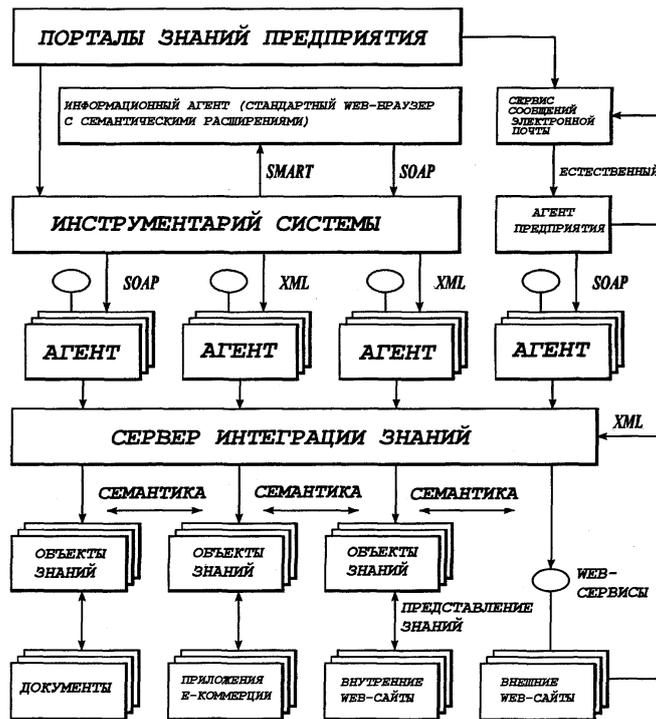
Фиг. 5

Уровень платформы	Современная сеть WEB	Информационная Нервная Система
Источник структурированной информации	Обрабатываются индивидуально	Обрабатываются регулярно: как источник метаданных и семантика
Источник неструктурированной информации	Обрабатываются индивидуально	Обрабатываются регулярно: как источник метаданных и семантика
Индексирование информации	На основе ключевых слов; поисковые машины не понимают отыскиваемые документы; семантики нет	На основе понятий; выделяет семантические метаданные из неструктурированной информации
Индексация и классификация знаний	Нет	Семантически кодирует и обобщает информацию из структурированных и неструктурированных источников
Представление знаний	Нет	Создает и поддерживает самовосстанавливаемую семантическую сеть объектов знаний
Онтология и логический вывод знания	Нет	Логически выводит новые связи и свойства в семантической сети
Логика	В основном база данных. Позволяет программировать поиск, правила, просмотр, запуск и т.д.	База знаний. Будет обеспечивать те же возможности программирования, но на уровне семантики
Приложение	Серверные сценарии, управляющие e-коммерцией; приложения, основанные на пользовательском вводе; Приложения ограничены представлением информации на основе пользовательского ввода	Серверные сценарии, программирующие базу знаний. Могут динамически генерировать объекты знания на основе пользовательского ввода. Сценарии могут включать семантические команды для извлечения уведомлений и логики. На этом уровне находится Интеллектуальный Агент для обработки семантического пользовательского ввода (подобно естественному языку)
Представление	Создается автором или динамически генерируется на сервере. Семантика теряется на сервере.	Полностью настраивается (посредством XML и XSLT); динамически генерируется у клиента; семантика сохраняется

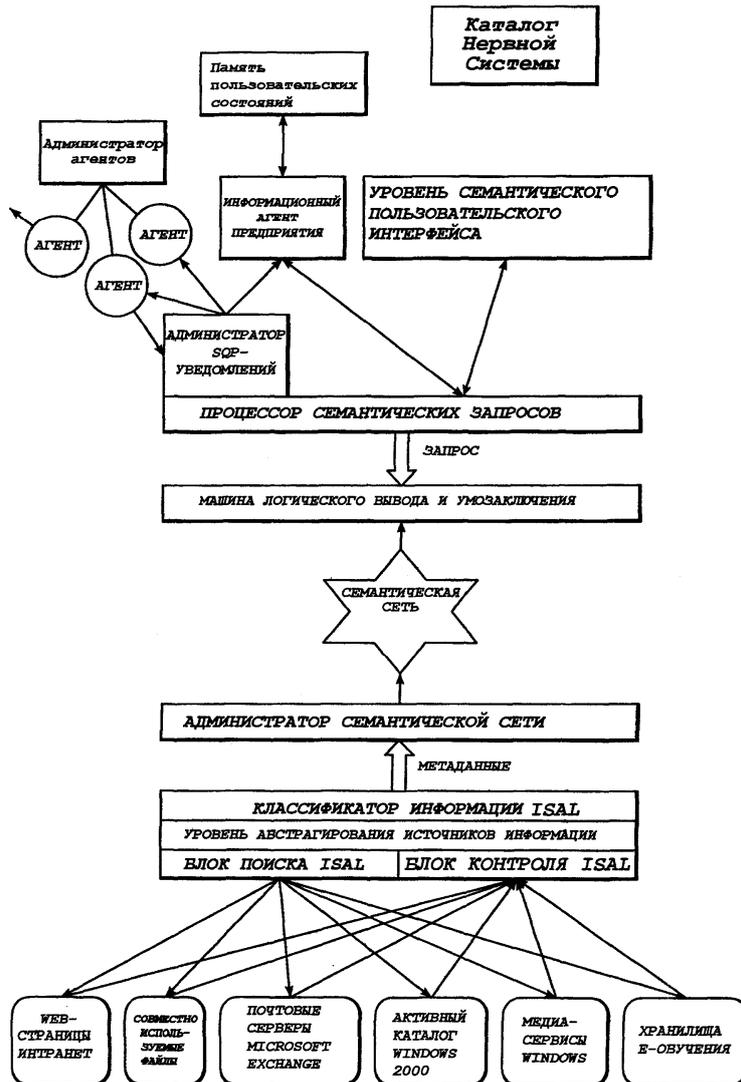
Фиг. 6



Фиг. 7



Фиг. 8



Фиг. 9

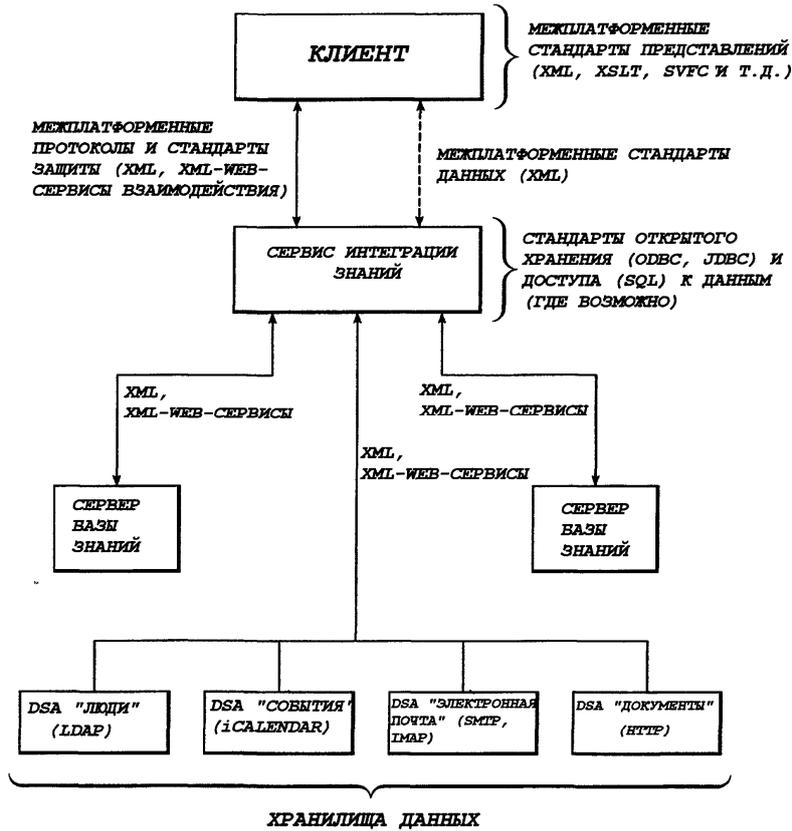
ИНФОРМАЦИОННЫЕ ПОРТАЛЫ ПРЕДПРИЯТИЯ	
ОТДЕЛЬНЫЕ ПРИЛОЖЕНИЯ Е- КОММЕРЦИИ	
WEB-СЕРВИСЫ ПРИЛОЖЕНИЙ	
БИЗНЕС-ЛОГИКА	ПОИСК
ЛОГИЧЕСКИЙ УРОВЕНЬ ДАННЫХ	ИНДЕКСИРОВАНИЕ ИНФОРМАЦИИ
ИСТОЧНИКИ СТРУКТУРИ- РОВАННОЙ ИНФОРМАЦИИ	ИСТОЧНИКИ НЕСТРУКТУРИ- РОВАННОЙ ИНФОРМАЦИИ

ПОРТАЛЫ ЗНАНИЙ ПРЕДПРИЯТИЯ	
ПРИЛОЖЕНИЯ Е-КОММЕРЦИИ НА ОСНОВЕ ИНТЕГРИРОВАННЫХ ЗНАНИЙ	
WEB-СЕРВИСЫ ПРИЛОЖЕНИЙ НА ОСНОВЕ ЗНАНИЙ (АГЕНТЫ, НАВИГАЦИЯ, СОЕДИНЕНИЯ, ОТКРЫТИЯ, СОВМЕСТНОЕ ИСПОЛЬЗОВАНИЕ)	
ЛОГИКА ЗНАНИЙ	СЕРВИС ЗАПРОСОВ ЗНАНИЙ
СЕРВИСЫ ОНТОЛОГИИ ЗНАНИЙ/ЛОГИЧЕСКОГО ВЫВОДА	
ПРЕДСТАВЛЕНИЕ ЗНАНИЙ	
ИНДЕКСАЦИЯ И КЛАССИФИКАЦИЯ ЗНАНИЙ	
ИСТОЧНИКИ СТРУКТУРИ- РОВАННОЙ ИНФОРМАЦИИ	ИСТОЧНИКИ НЕСТРУКТУРИ- РОВАННОЙ ИНФОРМАЦИИ

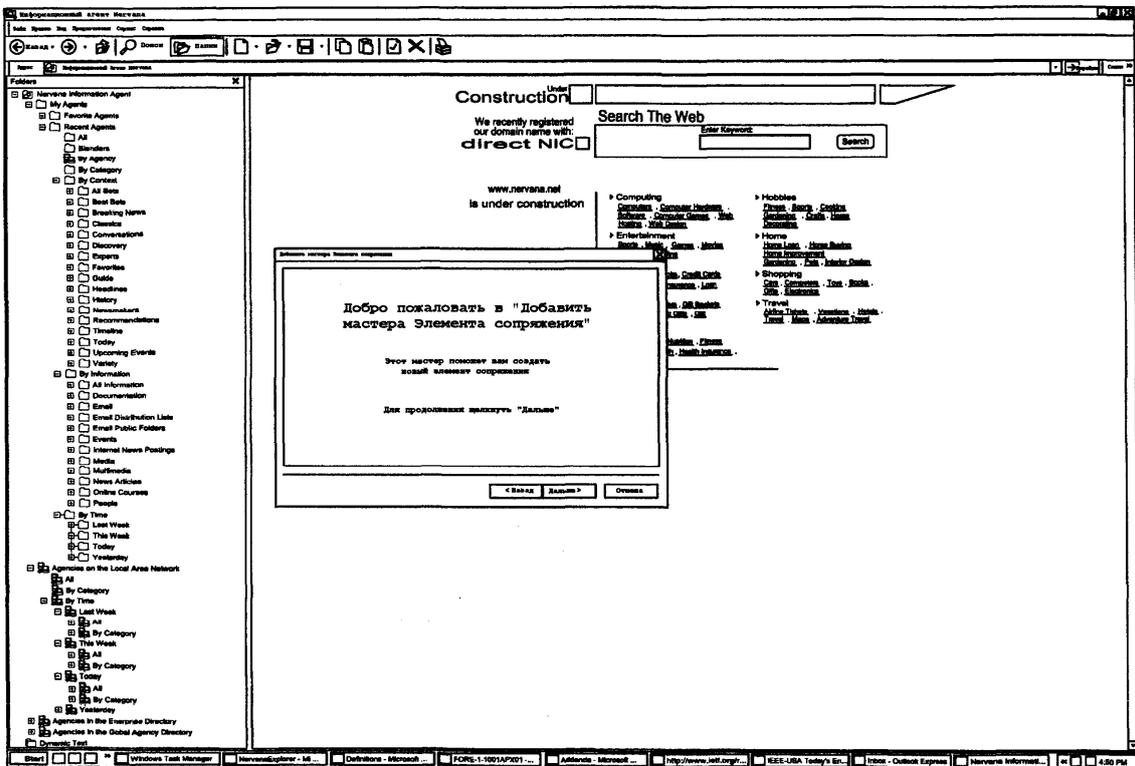
СОВРЕМЕННАЯ СЕТЬ WEB

**ИНФОРМАЦИОННАЯ НЕРВНАЯ
СИСТЕМА**

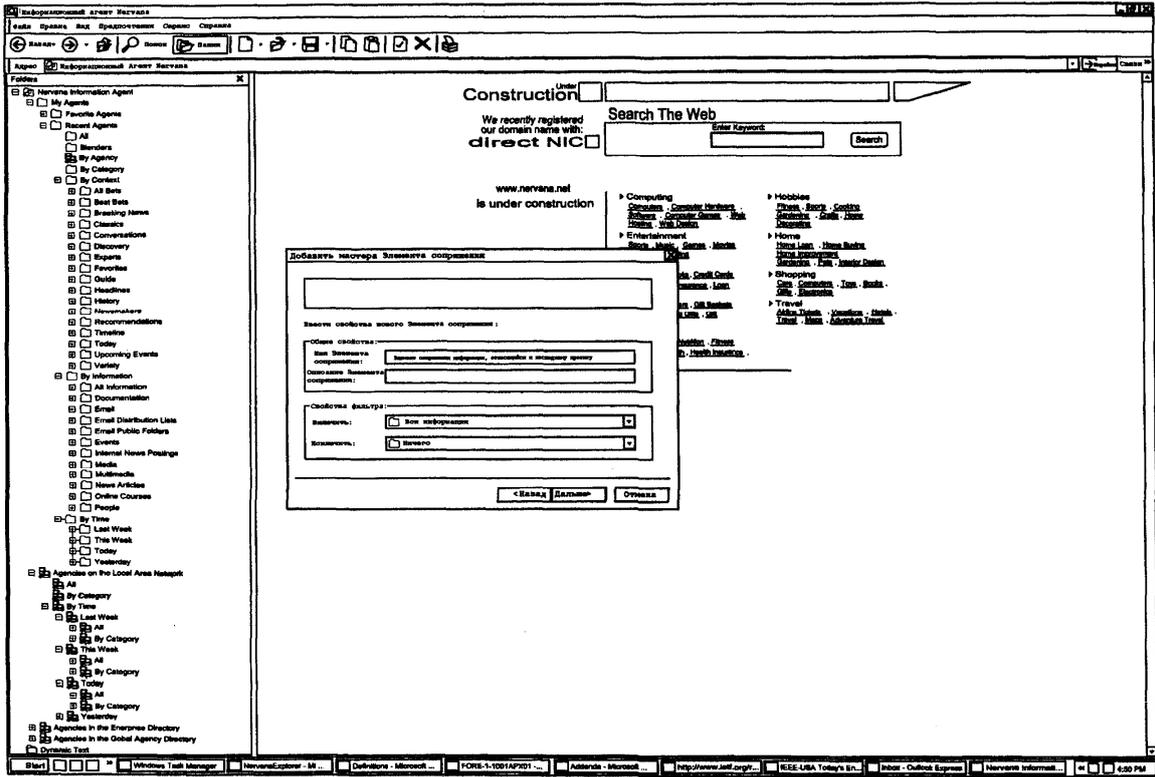
Фиг. 10



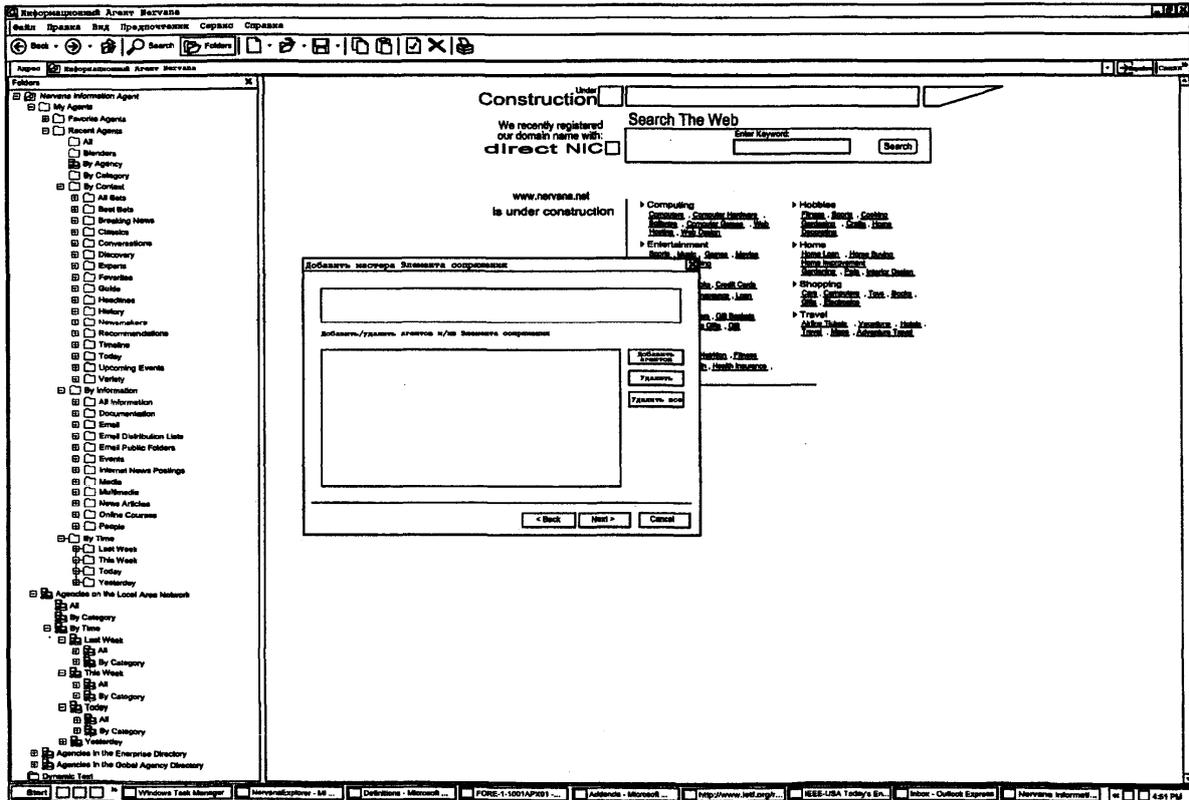
Фиг. 11



Фиг. 12



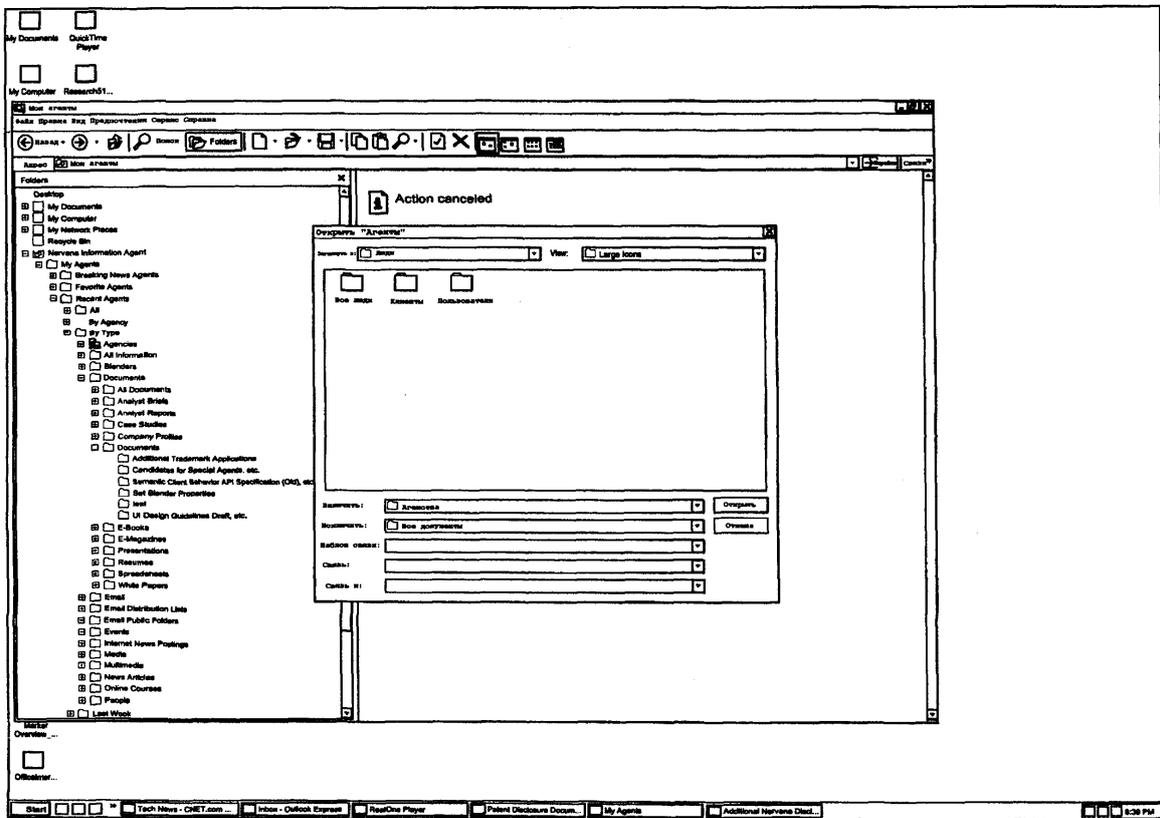
Фиг. 13



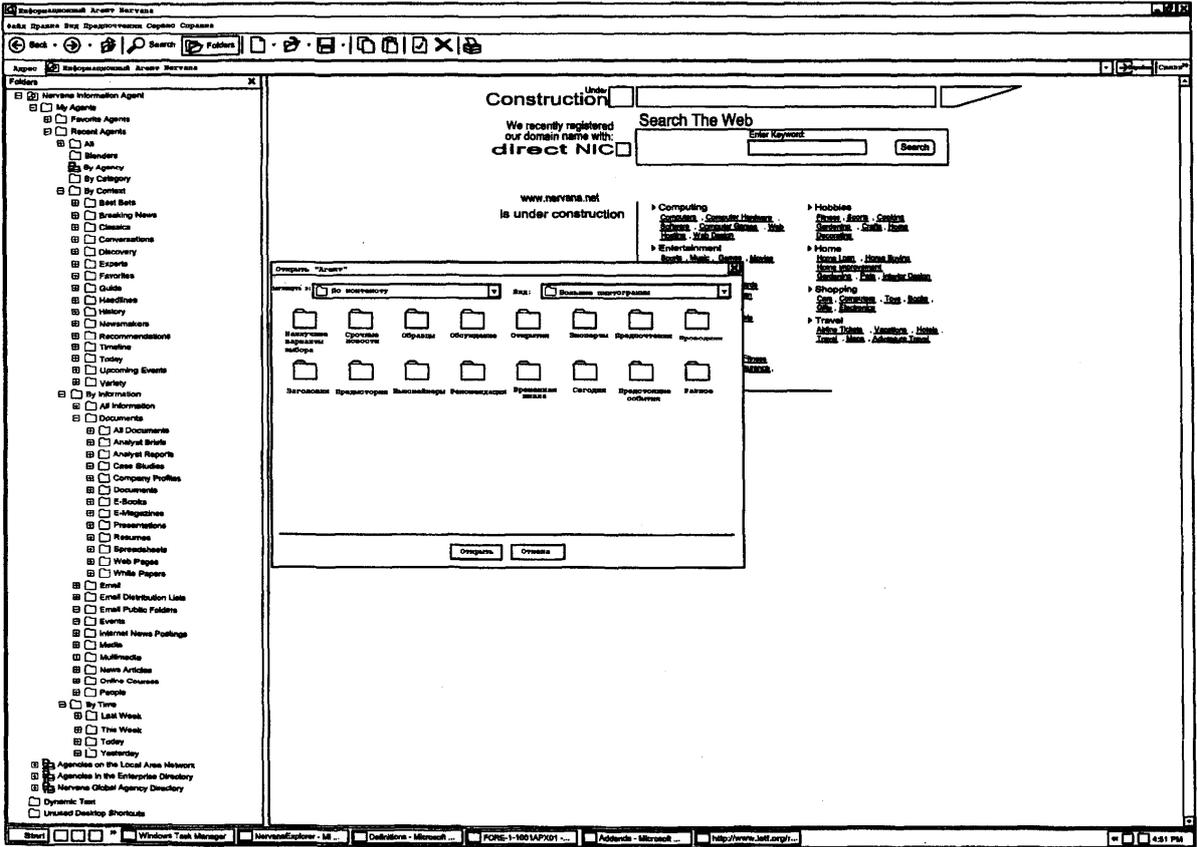
Фиг. 14

АГЕНТ НОВОСТЕЙ, ВЫЗЫВАЮЩИХ ПРЕРЫВАНИЕ: ДОКУМЕНТЫ REUTERS, ОТНОСЯЩИЕСЯ К [МОЯ СПЕЦИФИКАЦИЯ СОВРЕМЕННОГО ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА]	
НАЗВАНИЕ ОБЪЕКТА: [E-MAIL: ИДЕИ YUYING О СОВРЕМЕННОМ ПОЛЬЗОВАТЕЛЬСКОМ ИНТЕРФЕЙСЕ]	
СВЯЗЬ: ВСЯ ИНФОРМАЦИЯ, ОТНОСЯЩАЯСЯ К ЭТОМУ ЭЛЕМЕНТУ ↓	
ПОСЛЕДНИЙ ЭЛЕМЕНТ ПОСЛАН: 17 МАЯ, 12:13 PM [ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР]	
НАЧАЛО СЛЕДУЮЩЕГО СОВЕЩАНИЯ: 28 ИЮНЯ, 09:00 [ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР]	
ПОКАЗАТЬ ТОЛЬКО ЭЛЕМЕНТЫ СЛЕДУЮЩЕГО ТИПА:	<input type="text"/>
ПОКАЗАТЬ ТОЛЬКО ЭЛЕМЕНТЫ, ПОСЛАННЫЕ ПОСЛЕ:	<input type="text"/>
НАЙДЕНО ВСЕГО 50 ЭЛЕМЕНТОВ [ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР]	

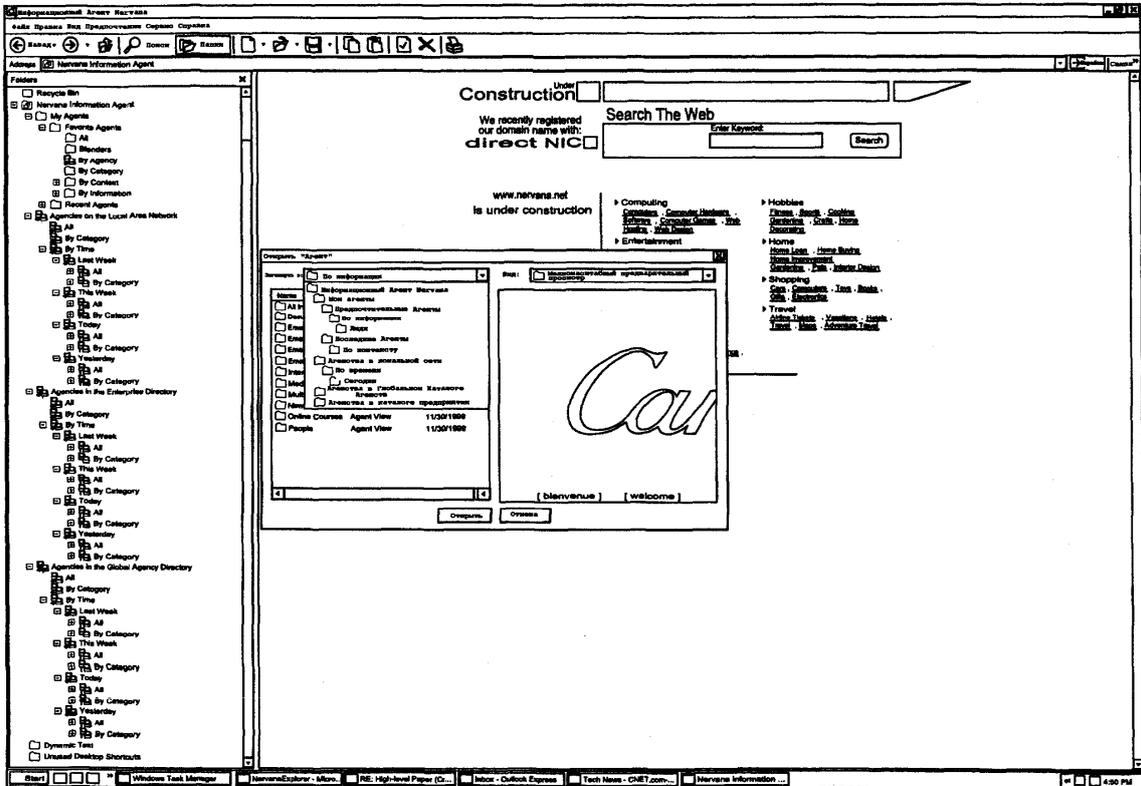
Фиг. 15



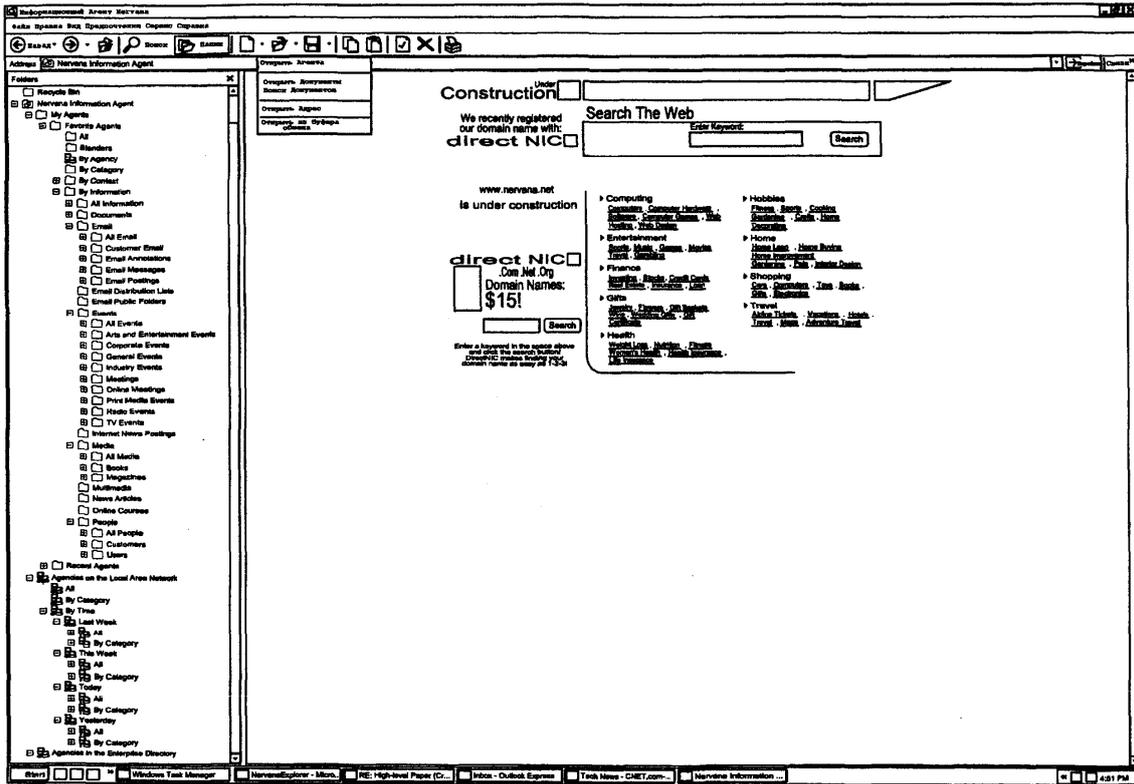
Фиг. 16



Фиг. 17



Фиг. 18



Фиг. 19

ИМЯ ПОЛЯ	ТИП
<i>ObjectID</i>	ЧИСЛО
<i>AgentTypeID</i>	ЧИСЛО
<i>AgentObjectTypeID</i>	ЧИСЛО
<i>AgentName</i>	ТЕКСТ
<i>AgentDescription</i>	ТЕКСТ
<i>AgentDefaultSkinUrl</i>	ТЕКСТ
<i>AgentQueryText</i>	ТЕКСТ
<i>AgentQueryTypeID</i>	ЧИСЛО

Фиг. 20

ИД ТИПА АГЕНТА	ЗНАЧЕНИЕ
<i>AGENTTYPEID_STANDARD</i>	1
<i>AGENTTYPEID_COMPOUND</i>	2
<i>AGENTTYPEID_BLENDER</i>	3
<i>AGENTTYPEID_SMART</i>	4
<i>AGENTTYPEID_SPECIAL</i>	5

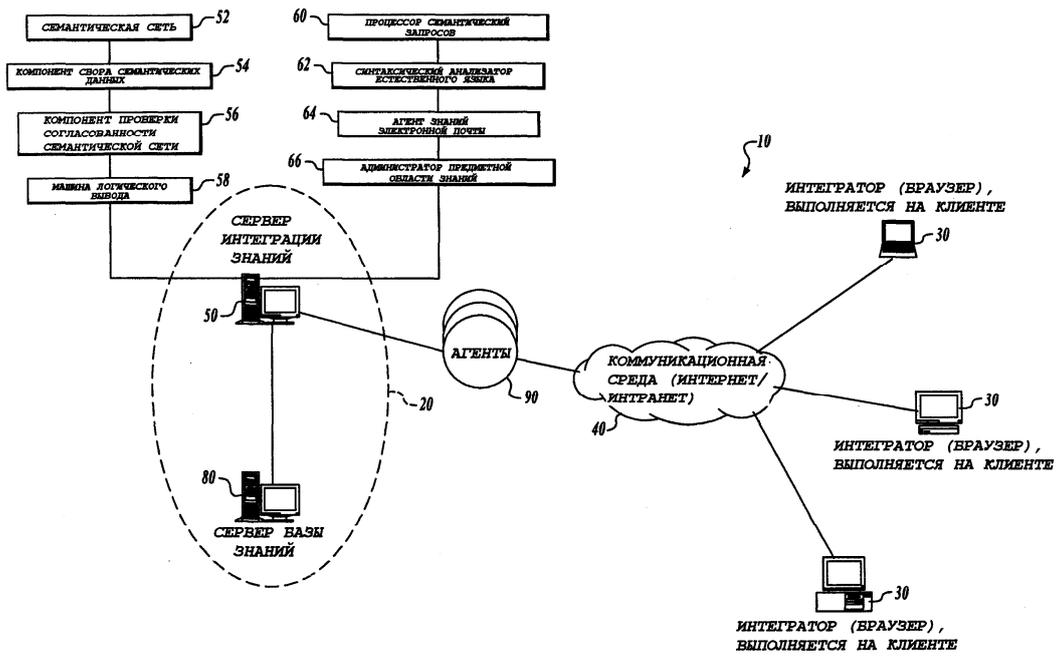
Фиг. 21

<i>AGENT QUERY TYPE ID</i>	ЗНАЧЕНИЕ
<i>AGENTQUERYTYPEID_SQL</i>	1

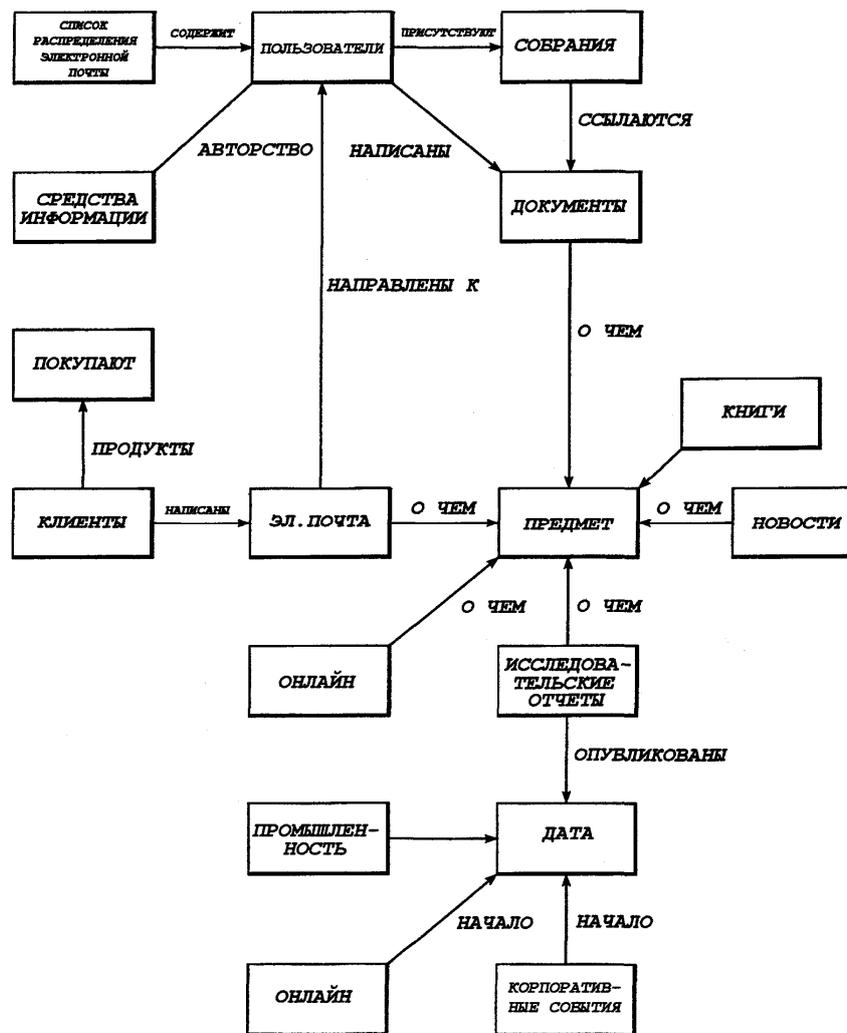
Фиг. 22

ИМЯ АГЕНТА	ТЕКСТ ЗАПРОСА АГЕНТА
<i>НОВОСТИ.ВСЕ</i>	ВЫБРАТЬ ИД ОБЪЕКТА ИЗ НОВОСТЕЙ
<i>НОВОСТИ.ТЕХНОЛОГИЯ.ВСЕ</i>	ВЫБРАТЬ ИД ОБЪЕКТА ИЗ НОВОСТЕЙ, ГДЕ ИД КАТЕГОРИИ="ТЕХНОЛОГИЯ.ВСЕ"

Фиг. 23



Фиг. 24



Фиг. 25

ИМЯ ПОЛЯ	ТИП
ObjectID	ЧИСЛЕННЫЙ
ObjectTypeID	ЧИСЛЕННЫЙ
Name	ТЕКСТОВЫЙ
Title	ТЕКСТОВЫЙ
Description	ТЕКСТОВЫЙ
URL	ТЕКСТОВЫЙ
CreationTime	ДАТА-ВРЕМЯ
LastModifiedTime	ДАТА-ВРЕМЯ
LastAccessTime	ДАТА-ВРЕМЯ
MajorVersion	ЧИСЛЕННЫЙ
MinorVersion	ЧИСЛЕННЫЙ
Attributes	ЧИСЛЕННЫЙ
SourceID	ЧИСЛЕННЫЙ
Custom	ТЕКСТОВЫЙ

Фиг. 26

ИМЯ ПОЛЯ	ТИП
SubjectID	ЧИСЛЕННЫЙ
SubjectTypeID	ЧИСЛЕННЫЙ
PredicateTypeID	ЧИСЛЕННЫЙ
ObjectID	ЧИСЛЕННЫЙ
ObjectTypeID	ЧИСЛЕННЫЙ
LinkScore	ЧИСЛЕННЫЙ
ReferenceDate	ДАТА-ВРЕМЯ

Фиг. 27

ИД ТИПА ПРЕДИКАТА	ЗНАЧЕНИЕ
PREDICATETYPEID_OFCATEGORY	1
PREDICATETYPEID_CREATOR	2
PREDICATETYPEID_ORGANIZER	3
PREDICATETYPEID_ATTENDEE	4
PREDICATETYPEID_CONTRIBUTOR	5
PREDICATETYPEID_APPROVER	6
PREDICATETYPEID_ATTACHEDTO	7
PREDICATETYPEID_SENTTO	8
PREDICATETYPEID_COPIEDTO	9
PREDICATETYPEID_BLINDCOPIEDTO	10
PREDICATETYPEID_REFERENCES	11
PREDICATETYPEID_SIMILARTO	12
PREDICATETYPEID_RELATEDTO	13
PREDICATETYPEID_EXPERTON	14
PREDICATETYPEID_REPORTSTO	15
PREDICATETYPEID_MEMBEROF	16
PREDICATETYPEID_INTERESTEDIN	17
PREDICATETYPEID_LIKELYTOBEINTERESTEDIN	18
PREDICATETYPEID_CONTAINSTAG	19
PREDICATETYPEID_CONTAINS	20
PREDICATETYPEID_ANNOTATES	21

Фиг. 28

СХЕМА ОБЪЕКТА "ПОЛЬЗОВАТЕЛЬ" ("ПЕРСОНА")	
ИМЯ ПОЛЯ	ТИП
FullName	ТЕКСТОВЫЙ
JobTitle	ТЕКСТОВЫЙ
Institution	ИД ИНСТИТУТА
BusinessPhone	ТЕКСТОВЫЙ
HomePhone	ТЕКСТОВЫЙ
BusinessFax	ТЕКСТОВЫЙ
MobilePhone	ТЕКСТОВЫЙ
Email	ТЕКСТОВЫЙ
DisplayAs	ТЕКСТОВЫЙ
HomeAddress	ТЕКСТОВЫЙ
BusinessAddress	ТЕКСТОВЫЙ
OtherAddress	ТЕКСТОВЫЙ
MailingAddressTypeID	ЧИСЛЕННЫЙ
WebPageAddress	ТЕКСТОВЫЙ
IMAddress	ТЕКСТОВЫЙ
OtherPhone	ДАТА-ВРЕМЯ
CarPhone	ДАТА-ВРЕМЯ

Фиг. 29А

OtherFax	ТЕКСТОВЫЙ
Pager	ЧИСЛЕННЫЙ
BusinessPhone2	ТЕКСТОВЫЙ
AssistantPhone	ТЕКСТОВЫЙ
CallbackPhone	ТЕКСТОВЫЙ
InstitutionPhone	ТЕКСТОВЫЙ
HomePhone2	ТЕКСТОВЫЙ
HomeFax	ТЕКСТОВЫЙ
ISDNPhone	ТЕКСТОВЫЙ
PrimaryPhone	ТЕКСТОВЫЙ
RadioPhone	ТЕКСТОВЫЙ
TelexPhone	ТЕКСТОВЫЙ
TTYPhone	ТЕКСТОВЫЙ
Department	ТЕКСТОВЫЙ
ManagerName	ТЕКСТОВЫЙ
Office	ТЕКСТОВЫЙ
AssistantName	ТЕКСТОВЫЙ
Profession	ТЕКСТОВЫЙ
Nickname	ТЕКСТОВЫЙ
SpouseName	ТЕКСТОВЫЙ
Birthday	ДАТА
Anniversary	ДАТА
OnlineConferenceServer	ТЕКСТОВЫЙ
OnlineConferenceEmailAlias	ТЕКСТОВЫЙ
InternetFreeBusyAddress	ТЕКСТОВЫЙ
Comments	ТЕКСТОВЫЙ

Фиг. 29В

ИД ТИПА ПОЧТОВОГО АДРЕСА	ЗНАЧЕНИЕ
MAILINGADDRESSTYPEID_HOME	1
MAILINGADDRESSTYPEID_BUSINESS	2
MAILINGADDRESSTYPEID_OTHER	3

Фиг. 30

СХЕМА ОБЪЕКТА "КАТЕГОРИЯ"	
ИМЯ ПОЛЯ	ТИП
CategoryUrl	ТЕКСТОВЫЙ
CategoryName	ТЕКСТОВЫЙ
CategoryDescription	ТЕКСТОВЫЙ
KBS Url	ТЕКСТОВЫЙ
Semantic Domain Name	ТЕКСТОВЫЙ

Фиг. 31

СХЕМА ОБЪЕКТА "ДОКУМЕНТ"	
ИМЯ ПОЛЯ	ИМЯ
Title	ТЕКСТОВЫЙ
Subject	ТЕКСТОВЫЙ
Author	ТЕКСТОВЫЙ
DocumentFormatTypeID	ЧИСЛЕННЫЙ
DocumentFormatVersion	ЧИСЛЕННЫЙ
Manager	ТЕКСТОВЫЙ
InstitutionID	ЧИСЛЕННЫЙ
DocumentCategory	ТЕКСТОВЫЙ
Keywords	ТЕКСТОВЫЙ
Summary	ТЕКСТОВЫЙ
Comments	ТЕКСТОВЫЙ
NumPages	ЧИСЛЕННЫЙ
NumParagraphs	ЧИСЛЕННЫЙ
NumLines	ЧИСЛЕННЫЙ
NumWords	ЧИСЛЕННЫЙ
CreationTime	ДАТА-ВРЕМЯ
LastModifiedTime	ДАТА-ВРЕМЯ
LastAccessedTime	ДАТА-ВРЕМЯ
LastSavedBy	ТЕКСТОВЫЙ
LanguageID	ЧИСЛЕННЫЙ
SourceUrl	ТЕКСТОВЫЙ

Фиг. 32

ИД ТИПА ПЕЧАТНЫХ ИСТОЧНИКОВ ИНФОРМАЦИИ	ЗНАЧЕНИЕ
DOCUMENTTYPEID_ANALYSTBRIEF	1
DOCUMENTTYPEID_ANALYSTREPORT	2
DOCUMENTTYPEID_CASESTUDY	3
DOCUMENTTYPEID_WHITEPAPER	4
DOCUMENTTYPEID_EBOOK	5
DOCUMENTTYPEID_EMAGAZINE	6
DOCUMENTTYPEID_NEWSARTICLE	7
DOCUMENTTYPEID_COMPANYPROFILE	8

Фиг. 33

ИД ТИПА ДОКУМЕНТА	ЗНАЧЕНИЕ
FORMATTYPEID_TEXT	1
FORMATTYPEID_HTML	2
FORMATTYPEID_ADOBEPDF	3
FORMATTYPEID_MSWORD	4
FORMATTYPEID_MSEXCEL	5
FORMATTYPEID_MSPowerPOINT	6
FORMATTYPEID_MSACCESS	7
FORMATTYPEID_MSPROJECT	8
FORMATTYPEID_MP3	9
FORMATTYPEID_WMA	10
FORMATTYPEID_RMA	11

Фиг. 34

СХЕМА ОБЪЕКТА "СООБЩЕНИЕ ЭЛ. ПОЧТЫ"	
ИМЯ ПОЛЯ	ТИП
From	ТЕКСТОВЫЙ
To	ТЕКСТОВЫЙ
Cc	ТЕКСТОВЫЙ
Bcc	ТЕКСТОВЫЙ
Subject	ТЕКСТОВЫЙ
EmailTypeID	ЧИСЛЕННЫЙ
ReceivedTime	ВРЕМЯ
Size	ЧИСЛЕННЫЙ
Priority	ЧИСЛЕННЫЙ
HeaderStatus	ЧИСЛЕННЫЙ
FlagStatus	БУЛЕВ (ЛОГИЧЕСКИЙ)
ContainsAttachment	БУЛЕВ (ЛОГИЧЕСКИЙ)
Sent	БУЛЕВ (ЛОГИЧЕСКИЙ)
Read	БУЛЕВ (ЛОГИЧЕСКИЙ)
FollowUpFlag	БУЛЕВ (ЛОГИЧЕСКИЙ)
CreatedTime	ДАТА-ВРЕМЯ
SentTime	ДАТА-ВРЕМЯ
Attachments	ТЕКСТОВЫЙ
SourceUri	ТЕКСТОВЫЙ

Фиг. 35

СХЕМА ОБЪЕКТА "СПИСОК РАСПРЕДЕЛЕНИЯ ЭЛ. ПОЧТЫ"	
ИМЯ ПОЛЯ	ТИП
DistributionListName	ТЕКСТОВЫЙ
DistributionListTypeID	ЧИСЛЕННЫЙ
Owner	ТЕКСТОВЫЙ
PublicAccess	БУЛЕВ (ЛОГИЧЕСКИЙ)
SourceUri	ТЕКСТОВЫЙ

Фиг. 36

СХЕМА ОБЪЕКТА "ОБЩЕДОСТУПНАЯ ПАПКА ЭЛ. ПОЧТЫ"	
ИМЯ ПОЛЯ	ТИП
PublicFolderName	ТЕКСТОВЫЙ
PublicFolderTypeID	ЧИСЛЕННЫЙ
Owner	ТЕКСТОВЫЙ
PublicAccess	БУЛЕВ (ЛОГИЧЕСКИЙ)
SourceUri	ТЕКСТОВЫЙ

Фиг. 37

ИД ТИПА СПИСКА РАСПРЕДЕЛЕНИЯ	ЗНАЧЕНИЕ
PUBLICFOLDERTYPEID_DISTRIBUTIONLIST	1
PUBLICFOLDERTYPEID_PUBLICFOLDER	2
PUBLICFOLDERTYPEID_NEWSGROUP	3

Фиг. 38

СХЕМА ОБЪЕКТА "СОБЫТИЕ"	
ИМЯ ПОЛЯ	ТИП
EventTitle	ТЕКСТОВЫЙ
EventID	ЧИСЛЕННЫЙ
EventTypeID	ЧИСЛЕННЫЙ
Status	ТЕКСТОВЫЙ
Location	ТЕКСТОВЫЙ
StartDate	ДАТА
StartTime	ВРЕМЯ
EndDate	ДАТА
EndTime	ВРЕМЯ
RequiredStaffing	ЧИСЛЕННЫЙ
Confirmed	БУЛЕВ (ЛОГИЧЕСКИЙ)
AvailableSpaces	ЧИСЛЕННЫЙ
CostPerPerson	ЧИСЛЕННЫЙ
CostCurrencyID	ЧИСЛЕННЫЙ
EventDescription	ТЕКСТОВЫЙ
Notes	ТЕКСТОВЫЙ
Organization	ТЕКСТОВЫЙ
EventOrganizer	ТЕКСТОВЫЙ
SourceUrl	ТЕКСТОВЫЙ

Фиг. 39

ИД ТИПА СОБЫТИЯ	ЗНАЧЕНИЕ
EVENTTYPEID_MEETING	1
EVENTTYPEID_CORPORATEEVENT	2
EVENTTYPEID_INDUSTRYEVENT	3
EVENTTYPEID_TVPROGRAM	4
EVENTTYPEID_RADIOPROGRAM	5
EVENTTYPEID_PRINTMEDIAPROGRAM	6
EVENTTYPEID_ONLINEMEETING	7
EVENTTYPEID_ARTSANDENTERTAINMENT	8

Фиг. 40

СХЕМА ОБЪЕКТА "ИСТОЧНИКИ ИНФОРМАЦИИ"	
ИМЯ ПОЛЯ	ТИП
Title	ТЕКСТОВЫЙ
PublishedDate	ДАТА
PublisherInstitutionID	ЧИСЛЕННЫЙ
Language	ЧИСЛЕННЫЙ
FormatTypeID	ЧИСЛЕННЫЙ
FormatVersion	ЧИСЛЕННЫЙ
ReportTypeID	ЧИСЛЕННЫЙ
NumPages	ЧИСЛЕННЫЙ
Abstract	ТЕКСТОВЫЙ
Author	ТЕКСТОВЫЙ
ProvidedBy	ТЕКСТОВЫЙ
SourceUrl	ТЕКСТОВЫЙ
Edition	ЧИСЛЕННЫЙ
ImageUrl	ТЕКСТОВЫЙ
PurchaseUrl	ТЕКСТОВЫЙ
RightsProtectedFlag	БУЛЕВ (ЛОГИЧЕСКИЙ)

Фиг. 41

ИД ТИПА ПЕЧАТНЫХ ИСТОЧНИКОВ ИНФОРМАЦИИ	ЗНАЧЕНИЕ
MEDIATYPEID_BOOK	1
MEDIATYPEID_MAGAZINE	2

Фиг. 42

Тип контейнера	Описание
Вся информация	Тип корневого объекта - Все типы объектов наследуют из этого типа
Элементы сопряжения	Тип комплексного объекта - составной тип, который включает объекты любого типа объекта. Элементы сопряжения могут включать другие элементы сопряжения
Документы	Документы
Электронная почта	Сообщения электронной почты
Списки распределения электронной почты	Списки распределения электронной почты
Общедоступные папки электронной почты	Общедоступные папки электронной почты
События	События
Новые статьи	Новые статья (например, публикации Reuters)
Отсылка новостей в Интернет	Отсылка новостей в Интернет (из новостных групп)
Онлайновые курсы обучения	Курсы е-обучения (из систем LMS), например SABA, DOCENT и т.д.
Источники информации	Источники информации, например, книги, журналы
Мультимедиа	Мультимедиа
Люди	Люди

Фиг. 43

Тип контейнера
Вся информация
Элементы сопряжения
Документы/комментарии аналитиков
Документы/отчеты аналитиков
Документы/анализ проблемы
Документы/профили компаний
Документы/документы
Документы/электронные книги
Документы/электронные журналы
Документы/белые книги
Эл. почта/клиентская Эл. почта
Эл. почта/аннотации Эл. почты
Эл. почта/сообщения Эл. почты
Эл. почта/отправка Эл. почты
Списки распределения Эл. почты
Общедоступные папки Эл. почты
События/события из мира искусства и развлечений
События/корпоративные события
События/общие события
События/отраслевые события
События/собрания
События/онлайнные собрания
События/события из источников информации
События/события радиовещания
События/события ТВ
Новостные статьи
Отправка новостей по Интернет
Онлайнные курсы обучения
Источники информации/книги
Источники информации/журналы
Мультимедиа
Люди/пользователи
Люди/клиенты

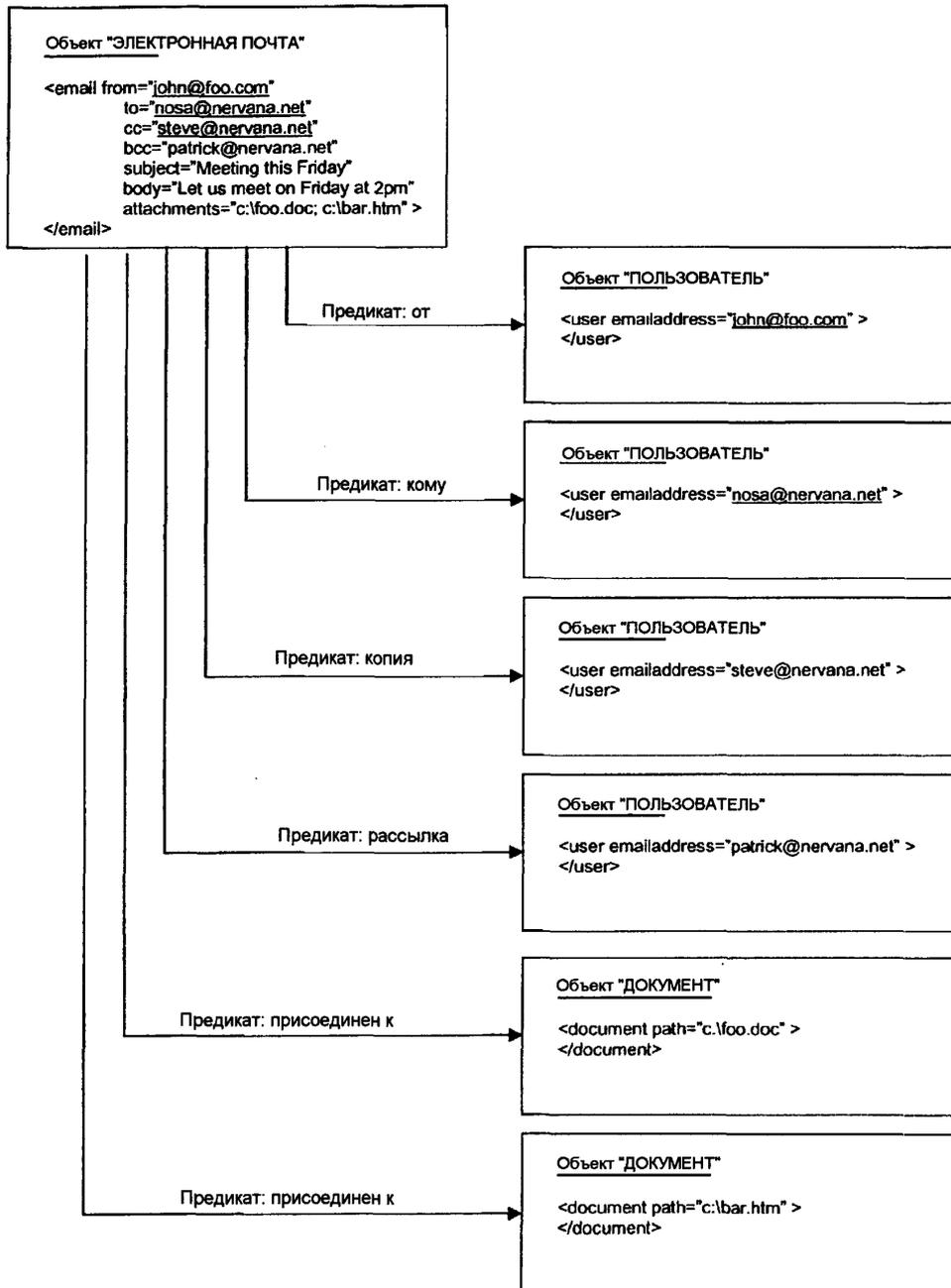
Фиг. 44

Тип контейнера	Префикс
Вся информация	Относится к
Вся информация	Возможно, относится к
Вся информация	Создано лицом по имени
Вся информация	Создано лицом с именем, содержащим текст
Вся информация	Создано лицом с адресом электронной почты
Вся информация	Возможно, создано лицом по имени
Вся информация	Возможно, создано лицом с именем, содержащим текст
Вся информация	Возможно, создано лицом с адресом электронной почты
Вся информация	Аннотировано лицом по имени
Вся информация	Аннотировано лицом с именем, содержащим текст
Вся информация	Аннотировано лицом с адресом электронной почты
Вся информация	С аннотациями, принадлежащими категории
Вся информация	Принадлежит к категории
Вся информация	Принадлежит к категориям, в которых лицо со следующим именем является экспертом
Вся информация	Принадлежит к категориям, в которых лицо с именем, содержащим следующий текст, является экспертом
Вся информация	Принадлежит к категориям, в которых лицо со следующим адресом электронной почты экспертом
Вся информация	С аннотациями, принадлежащими к категории
Вся информация	Аннотации, принадлежащие к категориям, в которых лицо со следующим именем является экспертом
Вся информация	Аннотации, принадлежащие к категориям, в которых лицо с именем, содержащим следующий текст, является экспертом
Вся информация	С аннотациями, принадлежащими к категориям, в которых лицо со следующим адресом электронной почты является экспертом
Вся информация	Создано в
Вся информация	Создано перед
Вся информация	Создано после
Вся информация	Послано в
Вся информация	Послано перед
Вся информация	Послано после
Вся информация	Послано впоследствии

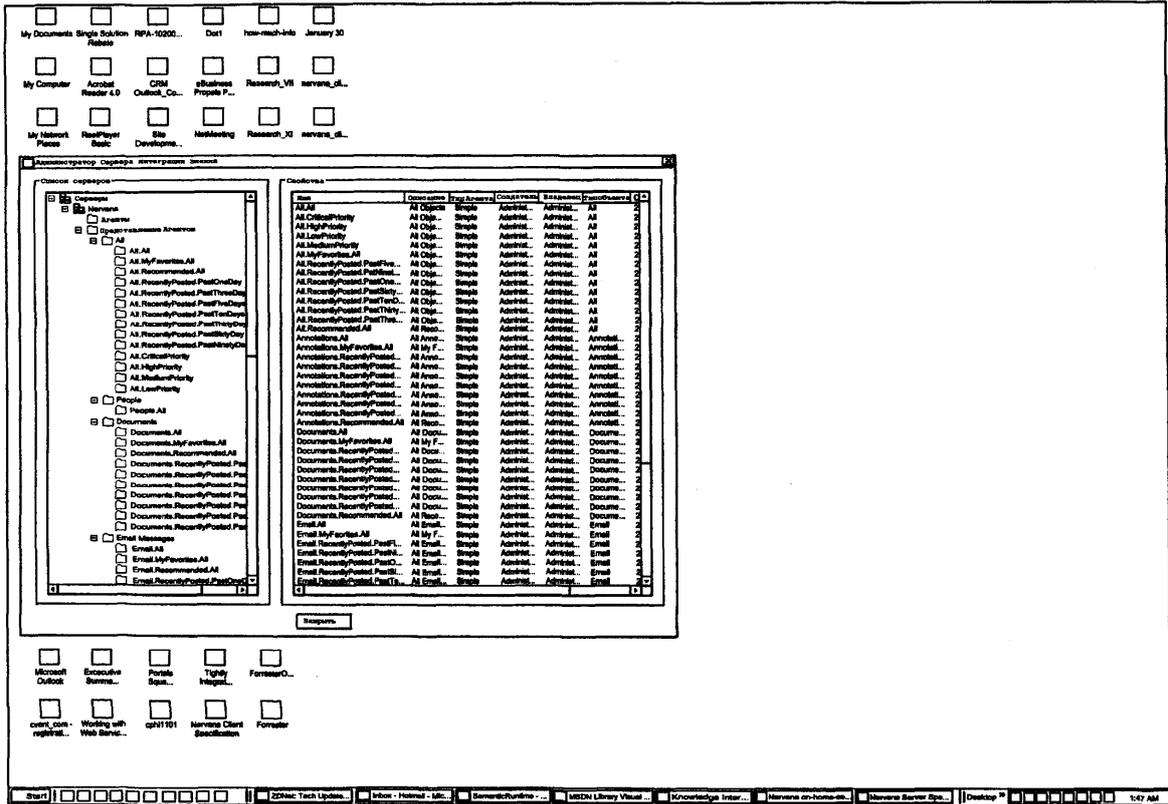
Фиг. 45А

Вся информация	С телом или названием, содержащим текст
Вся информация	С телом, содержащим текст
Вся информация	С названием, содержащим текст
Вся информация	С телом или названием с точным текстом
Вся информация	С телом с точным текстом
Вся информация	С названием с точным текстом
Эл. почта	Копия лицу с именем
Эл. почта	Копия лицу с именем, содержащим текст
Эл. почта	Копия лицу с адресом электронной почты
Эл. почта	Скрытая копия лицу с именем
Эл. почта	Скрытая копия лицу с именем, содержащим текст
Эл. почта	Скрытая копия лицу с адресом электронной почты
Эл. почта	Следующего приоритета
Эл. почта	Содержит приложения, принадлежащие к категории
Список распределения эл. почты	Содержит персону (ы) с именем,
Список распределения эл. почты	Содержит персону (ы) с именем, содержащим текст
Список распределения эл. почты	Содержит персону (ы) с адресом электронной почты
Событие	Начинается в
Событие	Началось в прошлом
Событие	Начинается в следующий
Событие	Заканчивается в
Событие	Закончилось в прошлом
Событие	Заканчивается в следующий
Событие	Повторяется каждый
Событие	Находится в месте с именем
Событие	Находится в месте с именем, содержащим текст
Событие	Организовано лицом с именем
Событие	Организовано лицом с именем, содержащим текст
Событие	Организовано лицом с адресом электронной почты
Событие	Посещается лицом с именем
Событие	Посещается лицом с именем, содержащим текст
Событие	Посещается лицом с адресом электронной почты
Люди	По имени
Люди	С именем, содержащим текст
Люди	С адресом электронной почты
Клиенты	По имени
Клиенты	С именем, содержащим текст
Клиенты	С адресом электронной почты

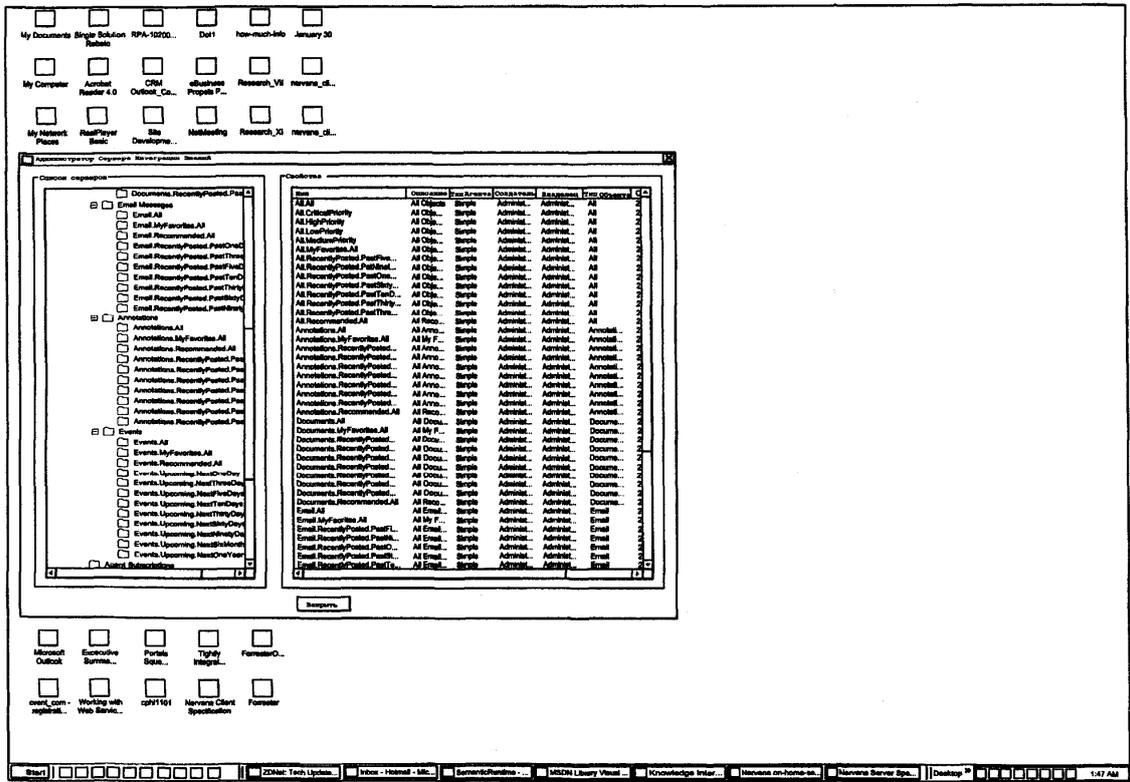
Фиг. 45B



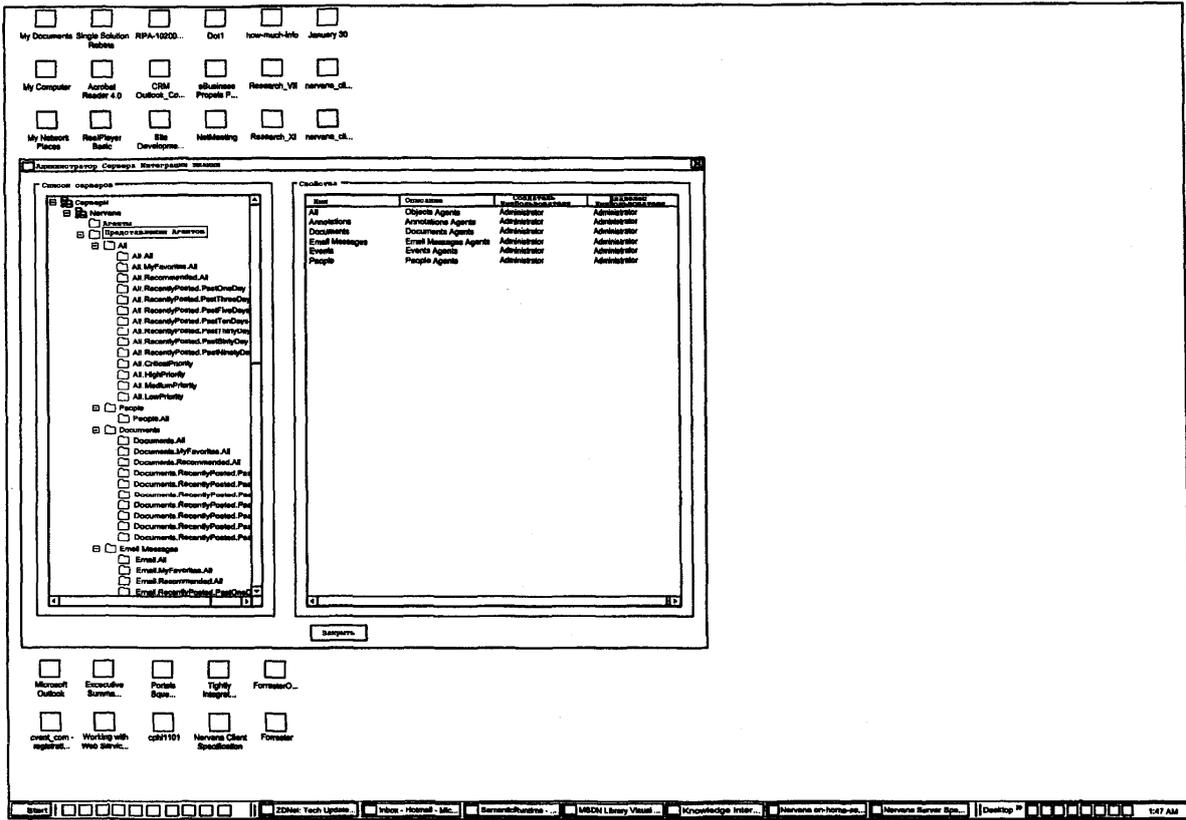
Фиг. 46



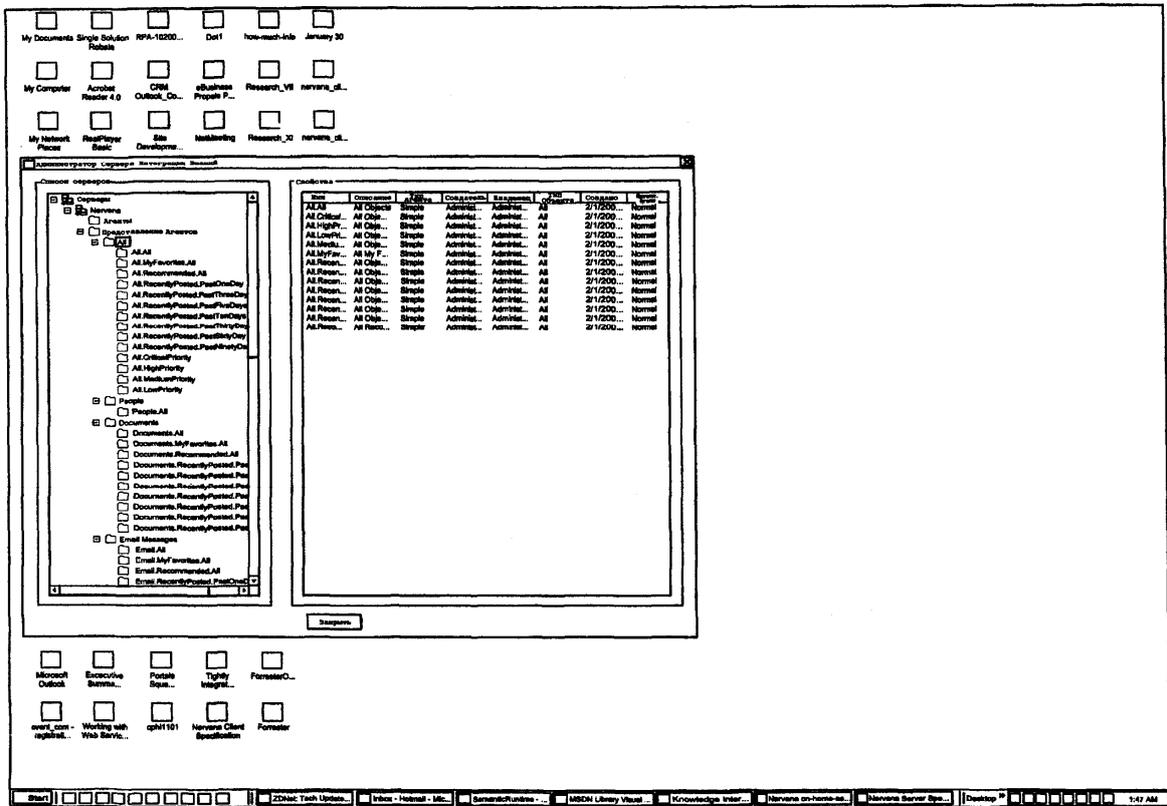
Фиг. 47



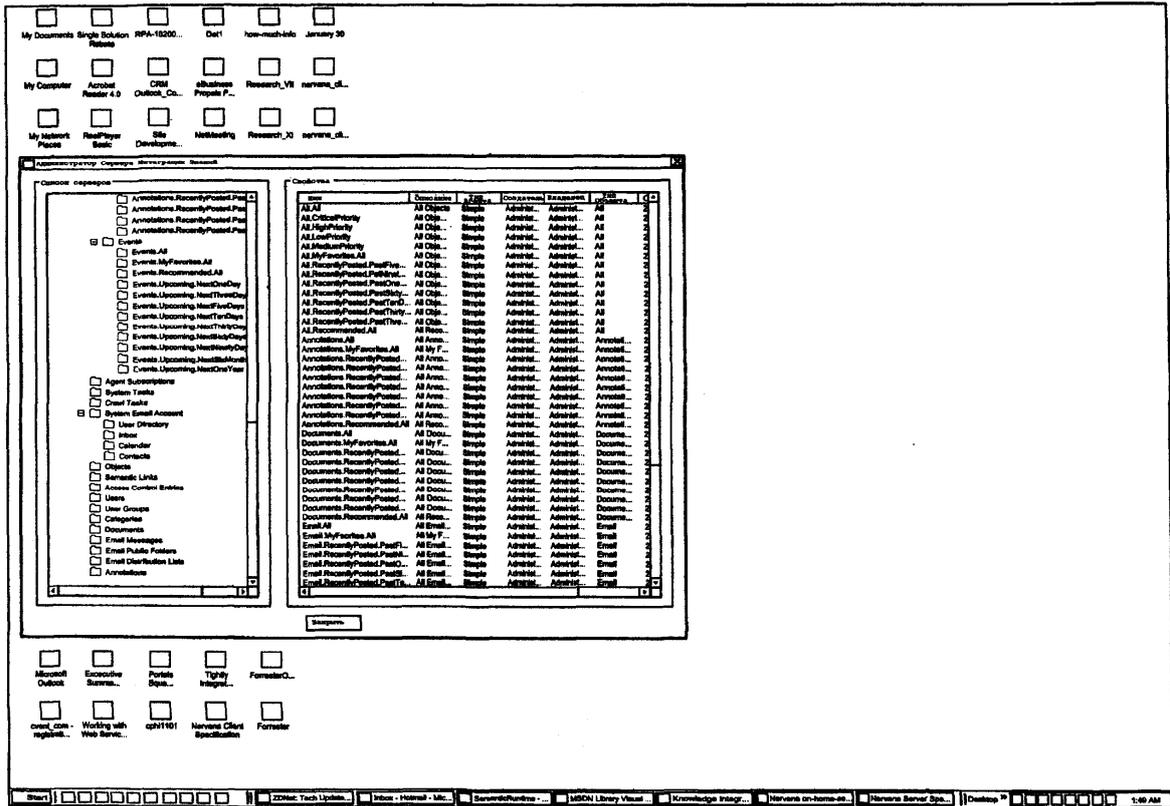
Фиг. 48



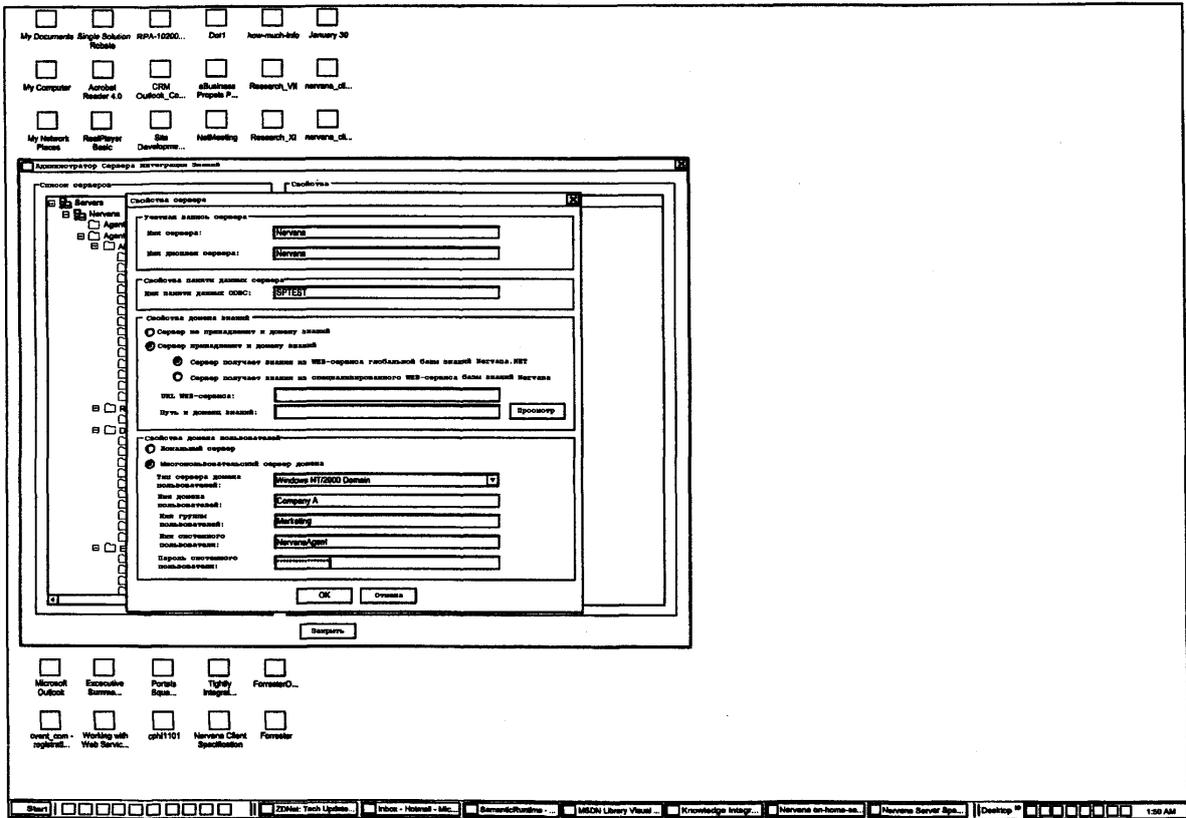
Фиг. 49



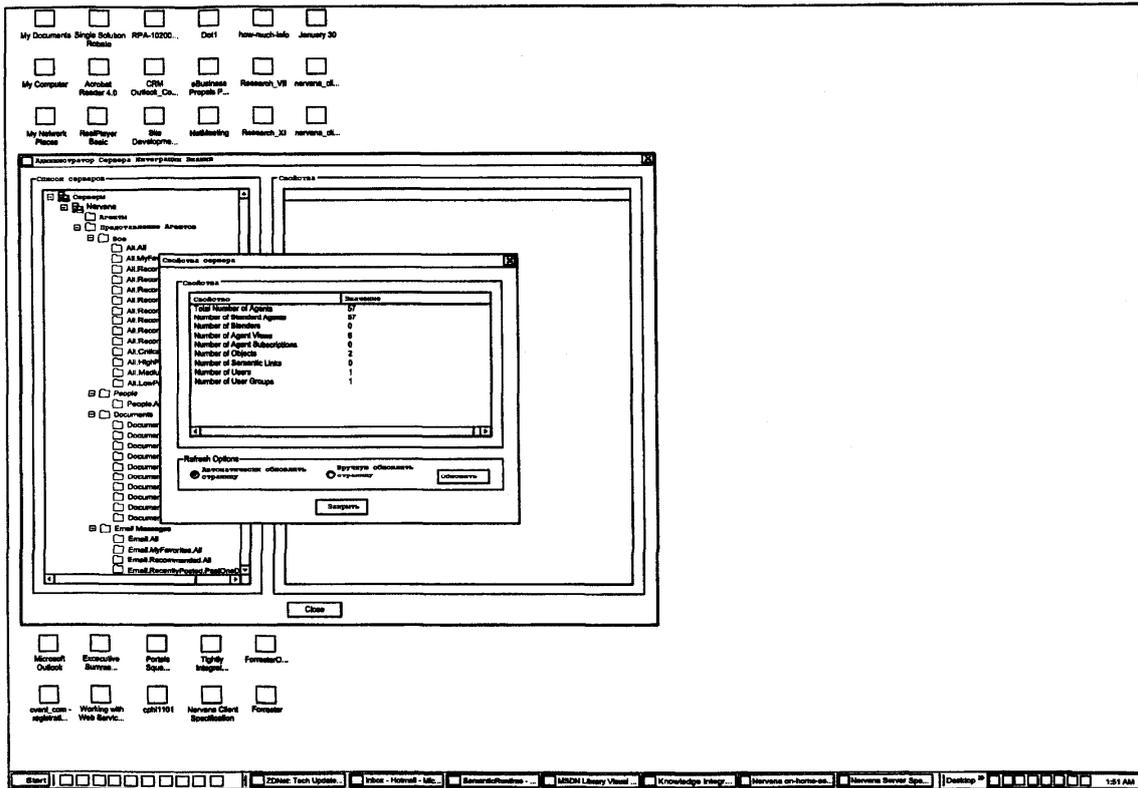
Фиг. 50



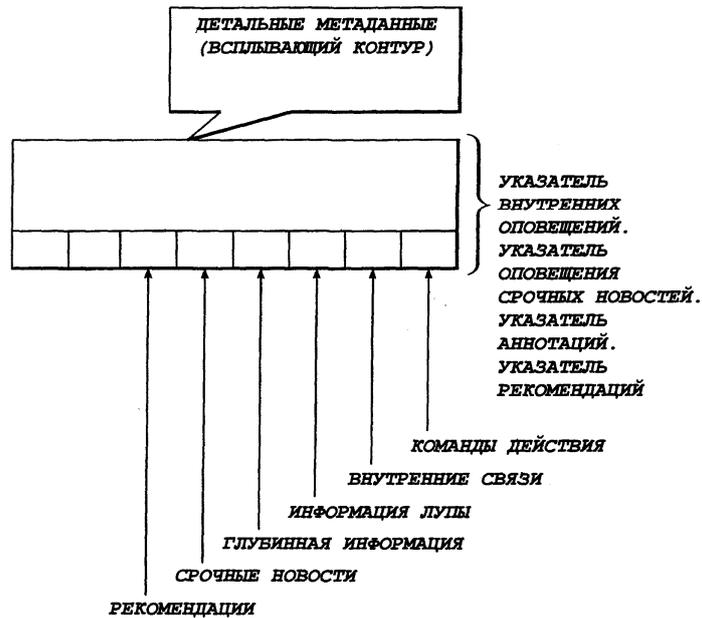
Фиг. 51



Фиг. 52



Фиг. 53



Фиг. 54

от ->[P]	
к ->[P]	
копия ->[P]	
СКРЫТАЯ КОПИЯ->[P]	
ПРИЛОЖЕНИЯ->[P]	ПРИЛОЖЕНИЯ 1 ->
	ПРИЛОЖЕНИЯ 1 ->
	ПРИЛОЖЕНИЯ 1 ->
	ПРИЛОЖЕНИЯ 1 ->
	ПРИЛОЖЕНИЯ 1 ->

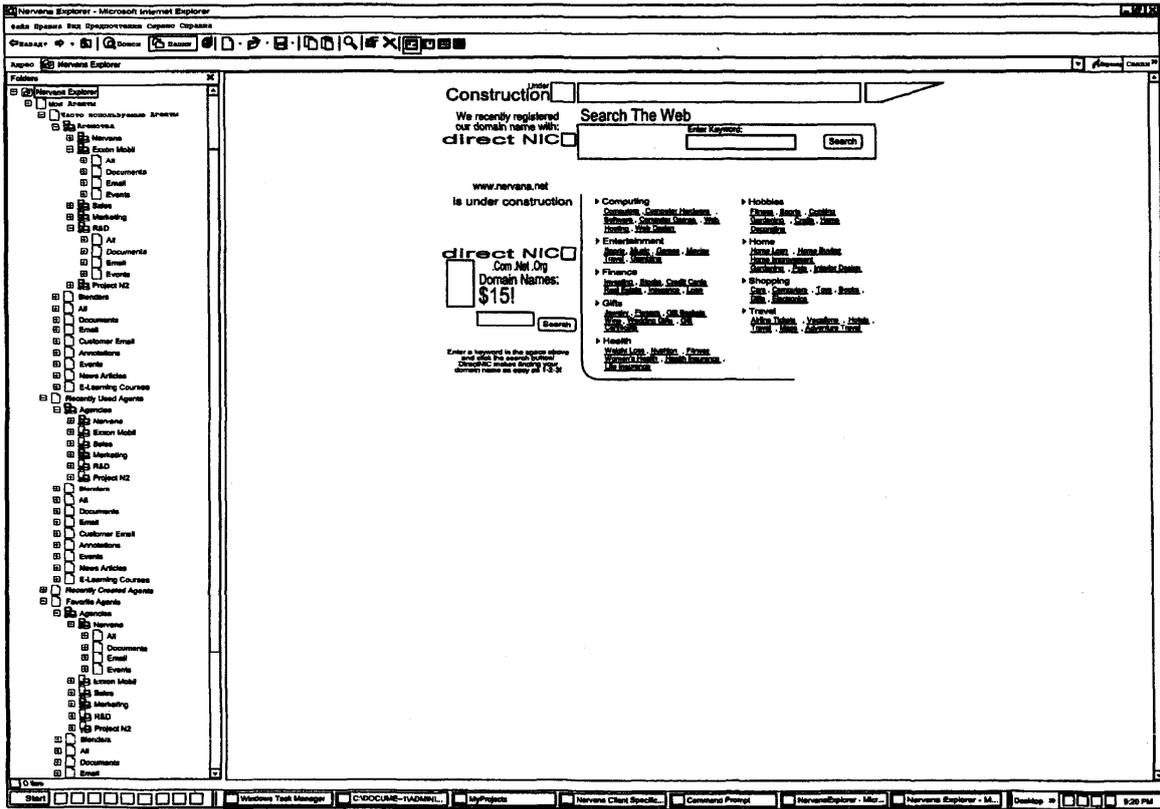
Фиг. 55

ОТКРЫТЬ
КОПИРОВАТЬ
АННОТИРОВАТЬ
СПЕЦИФИЧЕСКИЕ ДЛЯ ТИП КОМАНДЫ ДЕЙСТВИЯ (НАПРИМЕР, "ДОБАВИТЬ В КАЛЕНДАРЬ")

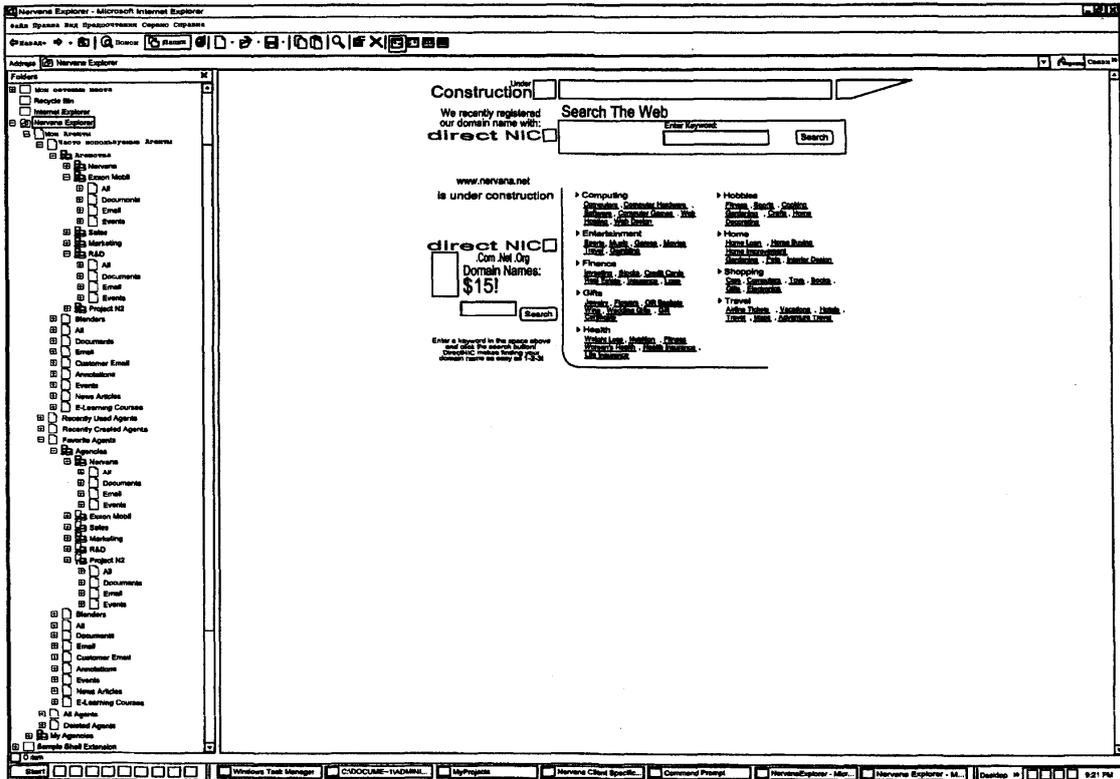
Фиг. 56

ШАБЛОН ИНФОРМАЦИОННОГО ЯДРА: [ШАБЛОН]
СТИВ ДЖАДКИНС НАИБОЛЕЕ ВЕРОЯТНО СОЗДАЛ ЭТОТ ДОКУМЕНТ [ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР]
ЭТОТ ДОКУМЕНТ КАТЕГОРИИ: [КАТЕГОРИЯ]
ЭТОТ ДОКУМЕНТ ИМЕЕТ 3 ВЕРОЯТНЫХ ЭКСПЕРТОВ [ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР]
СТИВ ДЖАДКИНС ИМЕЕТ 3 ПРЯМЫХ ОТЧЕТА [ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР]
СТИВ ДЖАДКИНС ПОСЛАЛ ВСЕГО 500 ЭЛЕМЕНТОВ НА ЭТО АГЕНСТВО [ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР]
СТИВ ДЖАДКИНС ПОСЛАЛ ВСЕГО 45 ИМЕЮЩИХ ОТНОШЕНИЕ ЭЛЕМЕНТОВ НА ЭТО АГЕНСТВО [ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР]
СТИВ ДЖАДКИНС ЯВЛЯЕТСЯ ЭКСПЕРТОМ ПО 78 ДОКУМЕНТАМ ЭТОГО АГЕНСТВА [ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР]
СТИВ ДЖАДКИНС ПРИСУТСТВУЕТ НА 12 ПРЕДСТОЯЩИХ СОБЫТИЯХ [ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР]

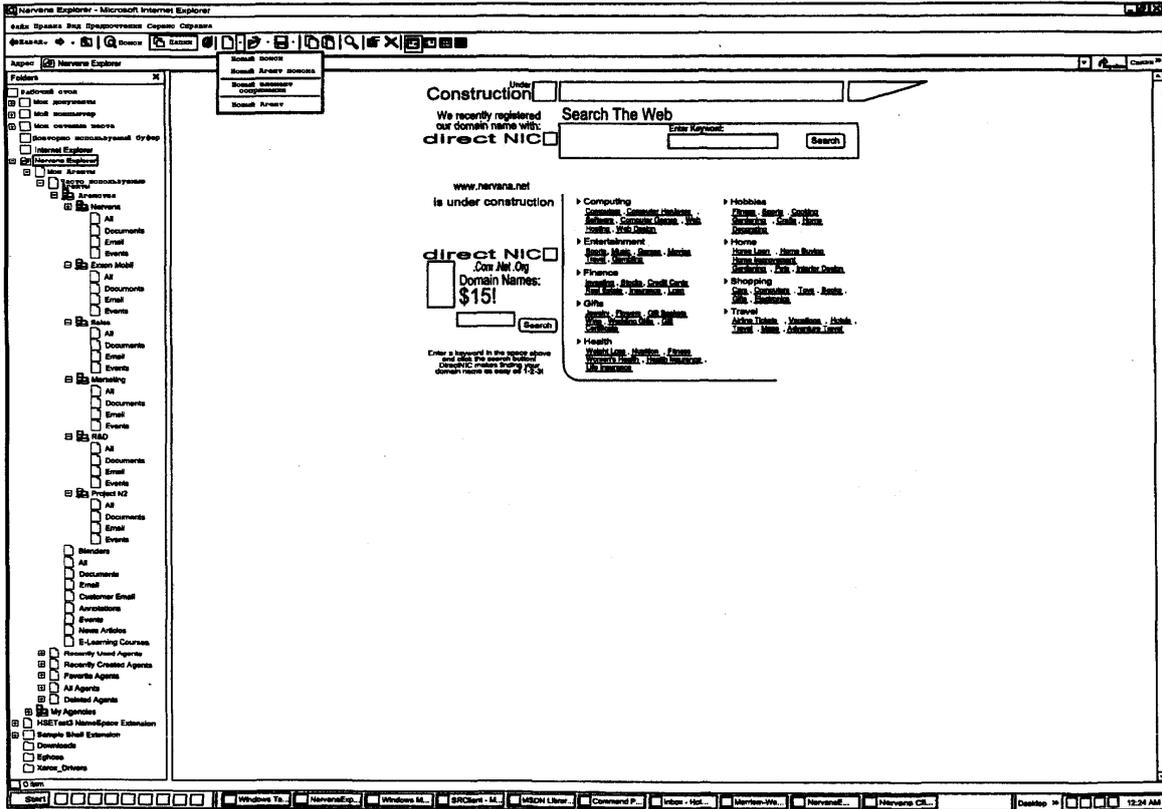
Фиг. 57



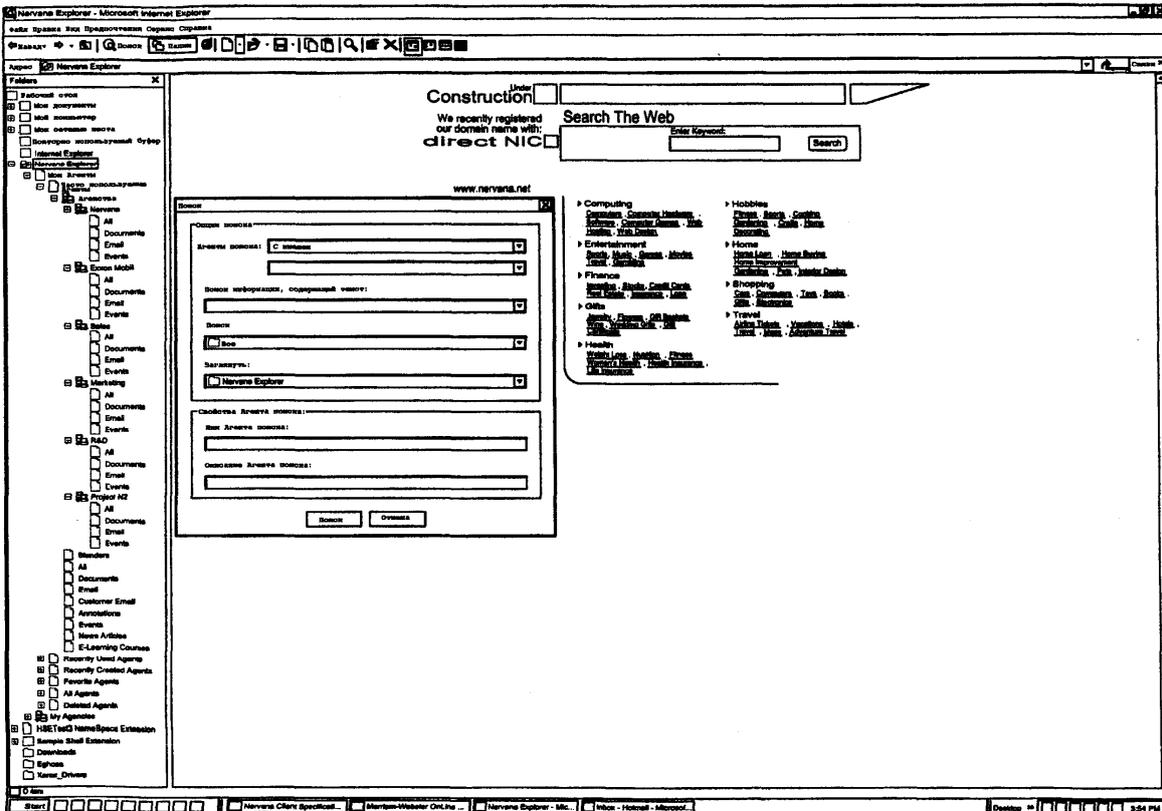
Фиг. 58



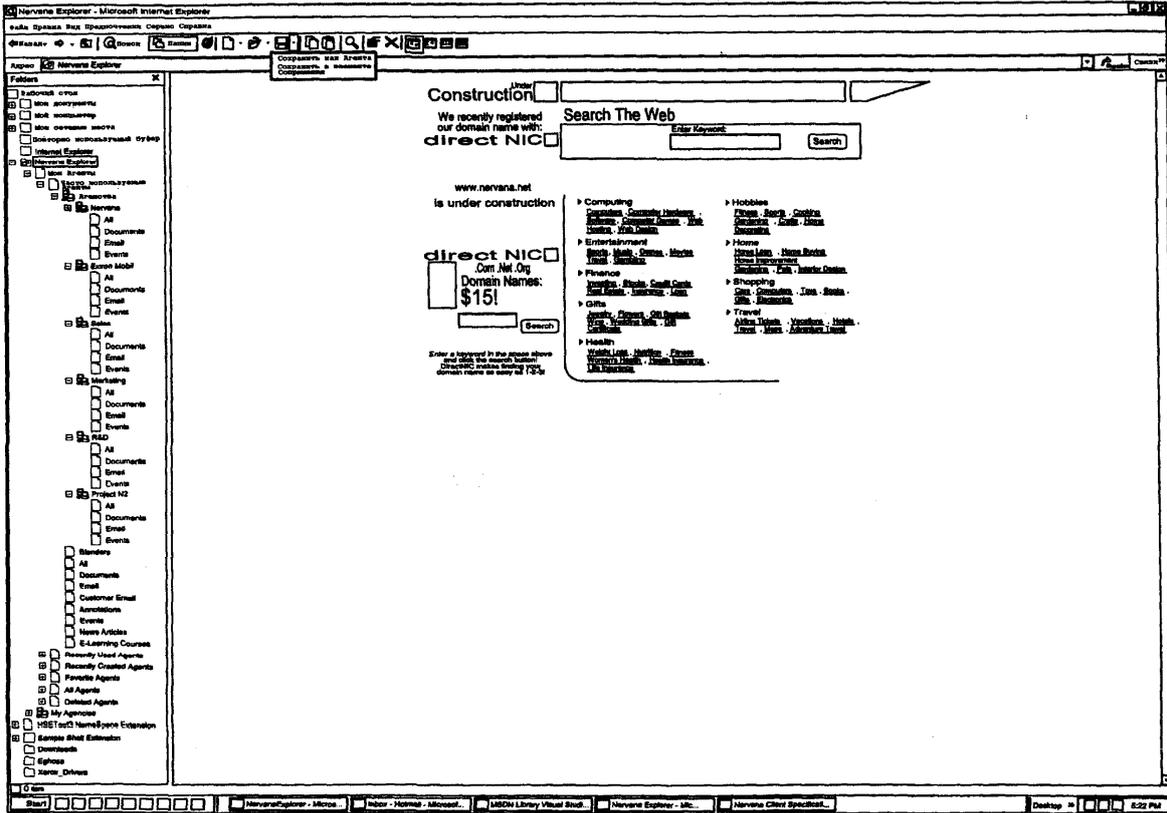
Фиг. 59



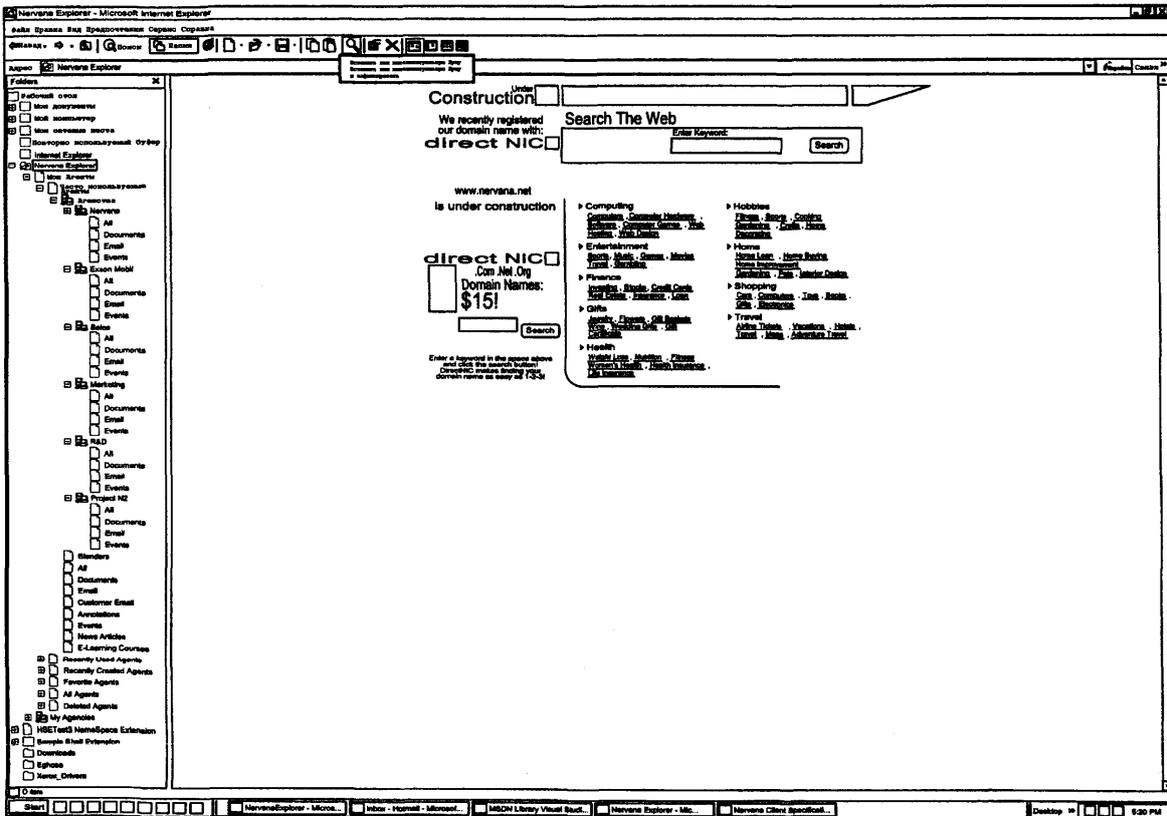
Фиг. 60



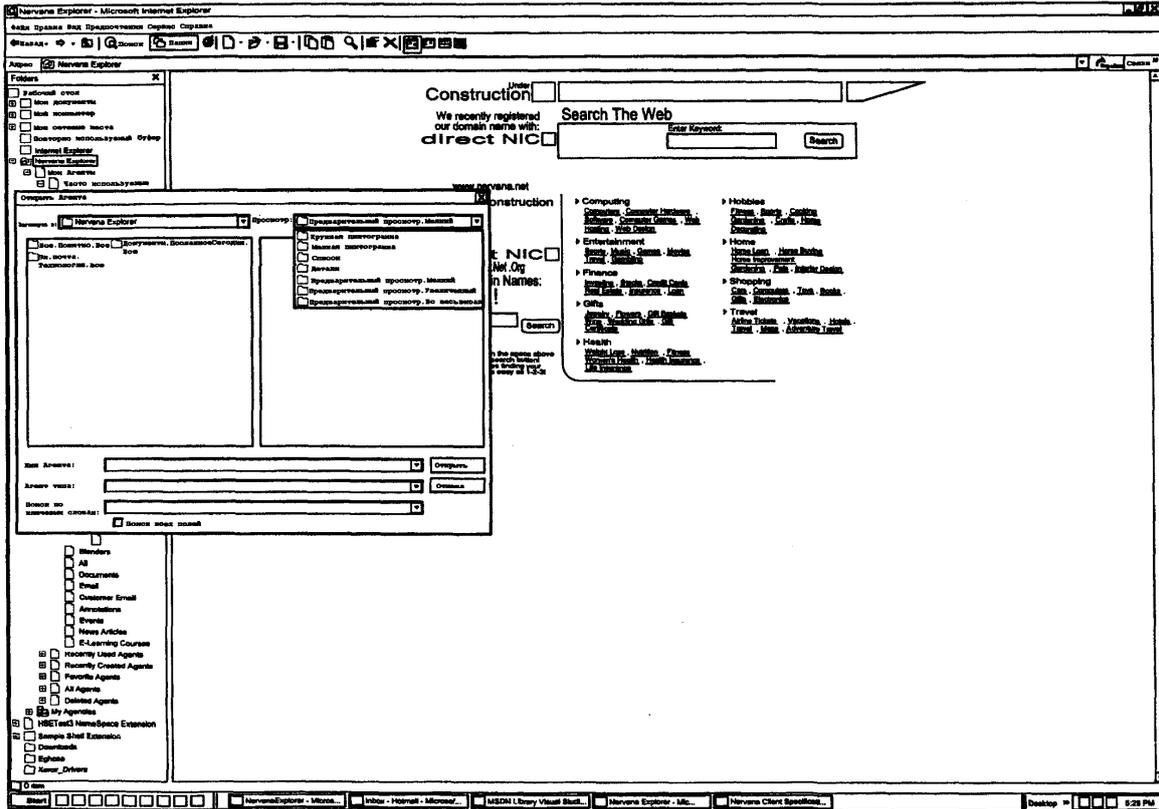
Фиг. 61



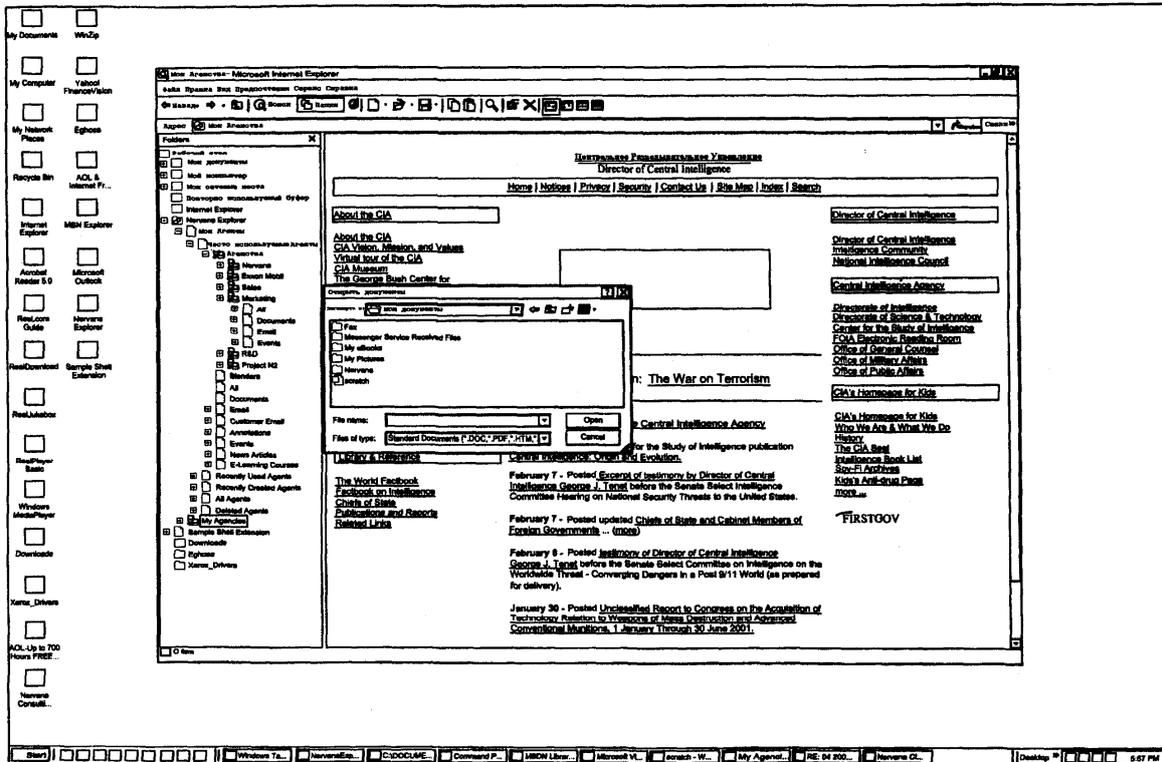
Фиг. 62



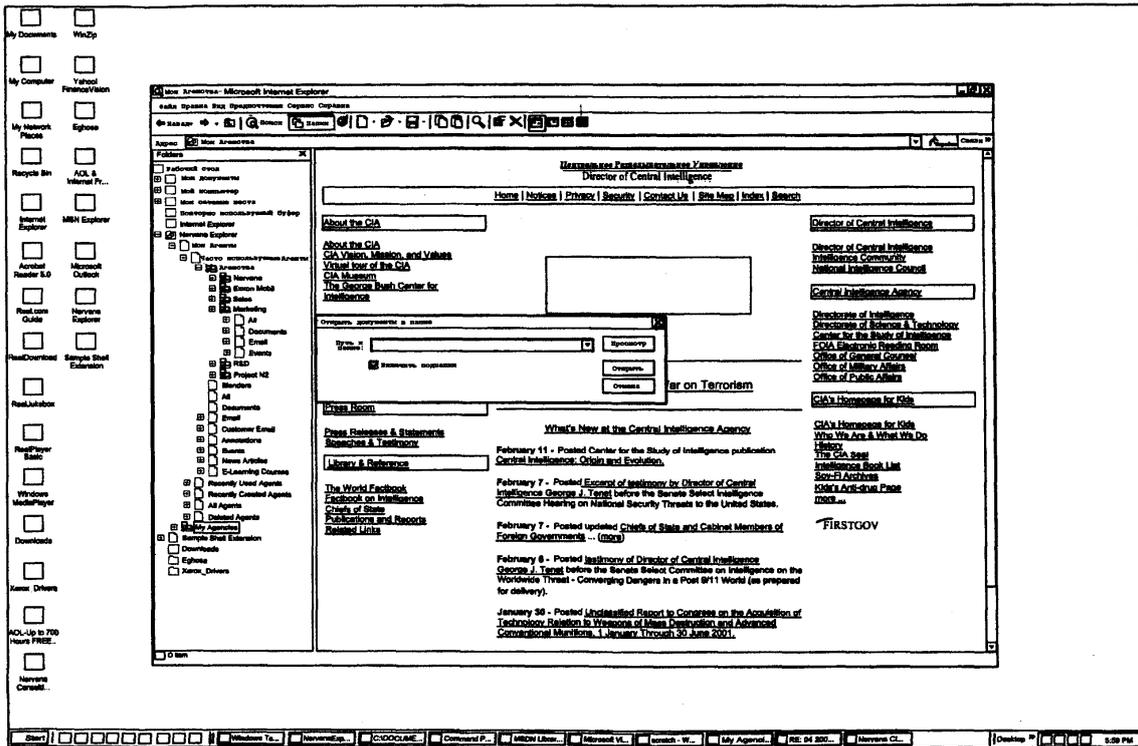
Фиг. 63



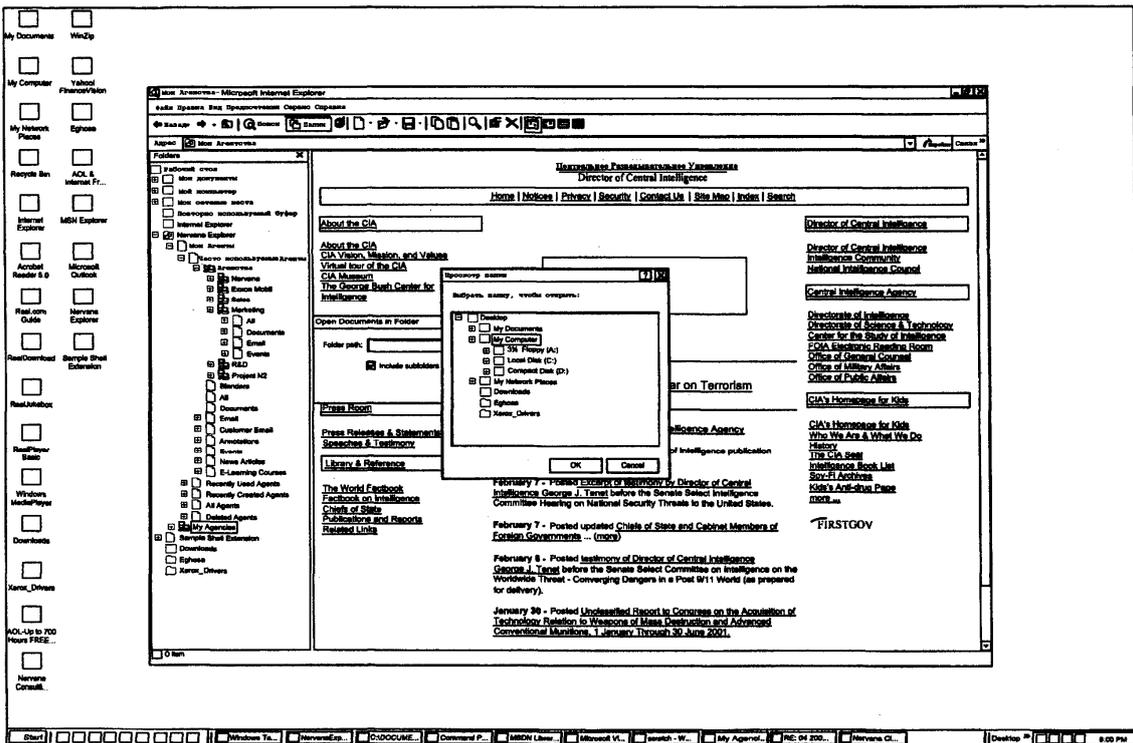
Фиг. 64



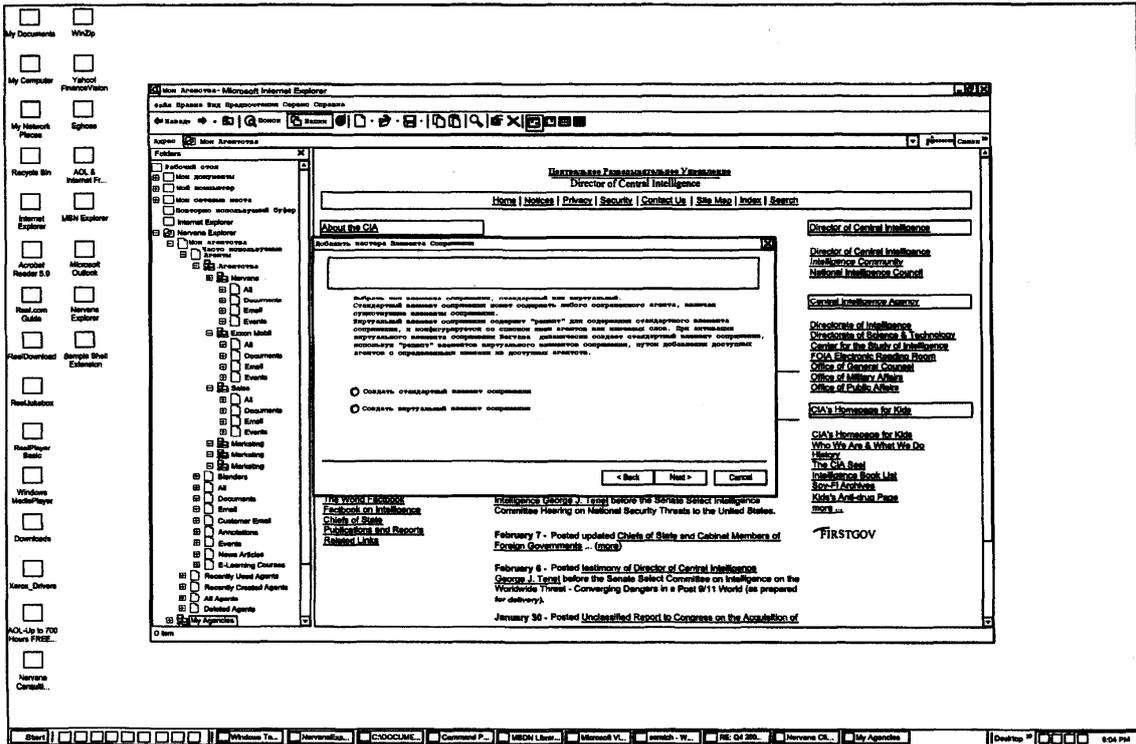
Фиг. 65



Фиг. 66



Фиг. 67



Фиг. 68

ЛУПА АГЕНТА: ДОКУМЕНТЫ REUTERS, ОТНОСЯЩИЕСЯ К [МОЯ СПЕЦИФИКАЦИЯ NERVANA UI]	
НАЗВАНИЕ ОБЪЕКТА:[ЭЛЕКТРОННАЯ ПОЧТА: ИДЕИ YUYING O NERVANAUI]	
СВЯЗЬ: ВСЯ ИНФОРМАЦИЯ, ОТНОСЯЩАЯСЯ К ЭТОМУ ЭЛЕМЕНТУ	
ПОСЛЕДНИЙ ЭЛЕМЕНТ ПОСЛАН: 17 МАЯ, 12:14 PM [ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР]	
СЛЕДУЮЩЕЕ СОБЫТИЕ НАЧНЕТСЯ: 28 ИЮНЯ, 09:00 AM [ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР]	
ПОКАЗАТЬ ТОЛЬКО ЭЛЕМЕНТЫ СЛЕДУЮЩЕГО ТИПА:	<input type="text"/>
ПОКАЗАТЬ ТОЛЬКО ЭЛЕМЕНТЫ, ПОСЛАННЫЕ ПОСЛЕДНИМИ:	<input type="text"/>
НАЙДЕНО ВСЕГО 50 ЭЛЕМЕНТОВ [ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР]	

Фиг. 69

НАЗВАНИЕ ЛУПЫ ОБЪЕКТА: [МОЯ СПЕЦИФИКАЦИЯ NERVANA UI]	
НАЗВАНИЕ АГЕНТА: ДОКУМЕНТЫ REUTERS, ОТНОСЯЩИЕСЯ К [МОЯ NERVANA UI]	
СВЯЗЬ: ВСЯ ИНФОРМАЦИЯ, ОТНОСЯЩАЯСЯ К ЭТОМУ ЭЛЕМЕНТУ	↓
ПОСЛЕДНИЙ ЭЛЕМЕНТ ПОСЛАН: 17 МАЯ, 12:14 РМ [ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР]	
СЛЕДУЮЩЕЕ СОБЫТИЕ НАЧНЕТСЯ: 28 ИЮНЯ, 09:00 АМ [ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР]	
ПОКАЗАТЬ ТОЛЬКО ЭЛЕМЕНТЫ СЛЕДУЮЩЕГО ТИПА:	<input type="text"/> ↓
ПОКАЗАТЬ ТОЛЬКО ЭЛЕМЕНТЫ, ПОСЛАННЫЕ ПОСЛЕДНИМИ:	<input type="text"/> ↓
НАЙДЕНО ВСЕГО 50 ЭЛЕМЕНТОВ [ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР]	

Фиг. 70

НАЗВАНИЕ ЛУПЫ ОБЪЕКТА: [МОЯ СПЕЦИФИКАЦИЯ NERVANA UI]	
НАЗВАНИЕ АГЕНТА: ДОКУМЕНТЫ REUTERS, ОТНОСЯЩИЕСЯ К [МОЯ СПЕЦИФИКАЦИЯ NERVANA UI]	
СВЯЗЬ: ВСЯ ИНФОРМАЦИЯ, ОТНОСЯЩАЯСЯ К ЭТОМУ ЭЛЕМЕНТУ	↓
ПОСЛЕДНИЙ ЭЛЕМЕНТ ПОСЛАН: 17 МАЯ, 12:14 РМ [ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР]	
СЛЕДУЮЩЕЕ СОБЫТИЕ НАЧНЕТСЯ: 28 ИЮНЯ, 09:00 АМ [ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР]	
ПОКАЗАТЬ ТОЛЬКО ЭЛЕМЕНТЫ СЛЕДУЮЩЕГО ТИПА:	<input type="text"/> ↓
ПОКАЗАТЬ ТОЛЬКО ЭЛЕМЕНТЫ, ПОСЛАННЫЕ ПОСЛЕДНИМИ:	<input type="text"/> ↓
НАЙДЕНО ВСЕГО 50 ЭЛЕМЕНТОВ [ПРЕДВАРИТЕЛЬНЫЙ ПРОСМОТР]	

Фиг. 71

НАЗВАНИЕ ЛУПЫ ОБЪЕКТА: [МОЯ СПЕЦИФИКАЦИЯ NERVANA UI]	
НАЗВАНИЕ ОБЪЕКТА: [ЭЛЕКТРОННАЯ ПОЧТА: ИДЕИ YUYING О NERVANA UI]	
СВЯЗЬ: МЕРА СВЯЗАННОСТИ СООТВЕТСТВЕННО	↓
АГЕНТСТВУ [ИМЯ АГЕНТСТВА]	
МЕРА СВЯЗАННОСТИ: 93%	

Фиг. 72

ТИП ЛУПЫ АГЕНТА	ТИП ОБЪЕКТА	ИЗОБРАЖЕНИЕ
ВСЯ ИНФОРМАЦИЯ	ДОКУМЕНТ	СВЯЗАННЫЕ ОБЪЕКТЫ НА АГЕНТЕ; ВОЗМОЖНО СВЯЗАННЫЕ ОБЪЕКТЫ НА АГЕНТЕ
ВСЯ ИНФОРМАЦИЯ	ЭЛ.ПОЧТА	СВЯЗАННЫЕ ОБЪЕКТЫ НА АГЕНТЕ; ВОЗМОЖНО СВЯЗАННЫЕ ОБЪЕКТЫ НА АГЕНТЕ
ВСЯ ИНФОРМАЦИЯ	СПИСОК РАСПРОСТРАНЕНИЯ ЭЛ.ПОЧТЫ	НЕТ
ВСЯ ИНФОРМАЦИЯ	ОБЩЕДОСТУПНАЯ ПАПКА ЭЛ.ПОЧТЫ	НЕТ
ВСЯ ИНФОРМАЦИЯ	СОБЫТИЕ	СВЯЗАННЫЕ ОБЪЕКТЫ НА АГЕНТЕ; ВОЗМОЖНО СВЯЗАННЫЕ ОБЪЕКТЫ НА АГЕНТЕ
ВСЯ ИНФОРМАЦИЯ	НОВОСТНАЯ СТАТЬЯ	СВЯЗАННЫЕ ОБЪЕКТЫ НА АГЕНТЕ; ВОЗМОЖНО СВЯЗАННЫЕ ОБЪЕКТЫ НА АГЕНТЕ
ВСЯ ИНФОРМАЦИЯ	НОВОСТНОЕ СООБЩЕНИЕ ИЗ ИНТЕРНЕТ	СВЯЗАННЫЕ ОБЪЕКТЫ НА АГЕНТЕ; ВОЗМОЖНО СВЯЗАННЫЕ ОБЪЕКТЫ НА АГЕНТЕ
ВСЯ ИНФОРМАЦИЯ	ОНЛАЙНОВЫЕ КУРСЫ ОБУЧЕНИЯ	СВЯЗАННЫЕ ОБЪЕКТЫ НА АГЕНТЕ; ВОЗМОЖНО СВЯЗАННЫЕ ОБЪЕКТЫ НА АГЕНТЕ
ВСЯ ИНФОРМАЦИЯ	ИСТОЧНИК ИНФОРМАЦИИ	СВЯЗАННЫЕ ОБЪЕКТЫ НА АГЕНТЕ; ВОЗМОЖНО СВЯЗАННЫЕ ОБЪЕКТЫ НА АГЕНТЕ
ВСЯ ИНФОРМАЦИЯ	МУЛЬТИМЕДИА	СВЯЗАННЫЕ ОБЪЕКТЫ НА АГЕНТЕ; ВОЗМОЖНО СВЯЗАННЫЕ ОБЪЕКТЫ НА АГЕНТЕ
ВСЯ ИНФОРМАЦИЯ	ПЕРСОНА	ОБЪЕКТЫ НА АГЕНТЕ, СОЗДАНИЕ ПЕРСОНОЙ; ОБЪЕКТЫ НА АГЕНТЕ, ВОЗМОЖНО СОЗДАНИЕ ПЕРСОНОЙ; ОБЪЕКТЫ НА АГЕНТЕ, АННОТИРОВАННЫЕ ПЕРСОНОЙ; ОБЪЕКТЫ НА АГЕНТЕ, НА КОТОРОМ ПЕРСОНА ЯВЛЯЕТСЯ ЭКСПЕРТОМ
ВСЯ ИНФОРМАЦИЯ	КЛИЕНТ	ОБЪЕКТЫ НА АГЕНТЕ, СОЗДАНИЕ ПЕРСОНОЙ; ОБЪЕКТЫ НА АГЕНТЕ, ВОЗМОЖНО СОЗДАНИЕ ПЕРСОНОЙ

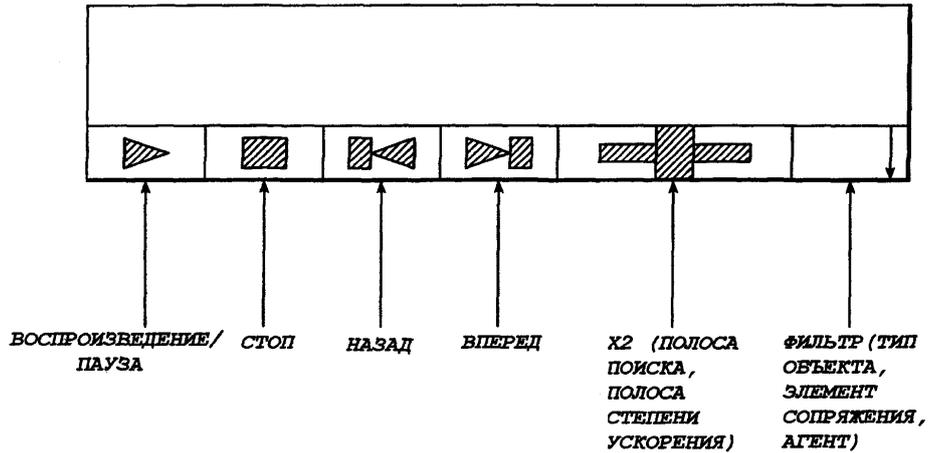
Фиг. 73

ТИП ЛУПЫ АГЕНТА	ТИП ОБЪЕКТА	ИЗОБРАЖЕНИЕ
ДОКУМЕНТ	ДОКУМЕНТ	ПРОЦЕНТ СХОДСТВА; СТОЛБЧАТАЯ ДИАГРАММА СХОДСТВА
ДОКУМЕНТ	ЭЛ.ПОЧТА	ПРОЦЕНТ СХОДСТВА; СТОЛБЧАТАЯ ДИАГРАММА СХОДСТВА
ДОКУМЕНТ	СПИСОК РАСПРОСТРАНЕНИЯ ЭЛ.ПОЧТЫ	ЧЛЕНЫ, СОЗДАВШИЕ ДОКУМЕНТ; ЧЛЕНЫ, КОТОРЫЕ МОГЛИ СОЗДАТЬ ДОКУМЕНТ; ЧЛЕНЫ, КОТОРЫЕ ЯВЛЯЮТСЯ ЭКСПЕРТАМИ ПО ДОКУМЕНТУ; БЫЛ ЛИ ДОКУМЕНТ КОГДА-ЛИВО ВКЛЮЧЕН В СПИСОК РАСПРЕДЕЛЕНИЯ
ДОКУМЕНТ	ОБЩЕДОСТУПНАЯ ПАПКА ЭЛ.ПОЧТЫ	СООБЩЕНИЯ ЭЛ.ПОЧТЫ В ОБЩЕДОСТУПНОЙ ПАПКЕ, КОТОРЫЕ ОТНОСЯТСЯ К ЭТОМУ ДОКУМЕНТУ; СООБЩЕНИЯ ЭЛ.ПОЧТЫ В ОБЩЕДОСТУПНОЙ ПАПКЕ, КОТОРЫЕ МОГУТ ОТНОСИТЬСЯ К ЭТОМУ ДОКУМЕНТУ
ДОКУМЕНТ	СОБЫТИЕ	ПРОЦЕНТ СХОДСТВА; СТОЛБЧАТАЯ ДИАГРАММА СХОДСТВА
ДОКУМЕНТ	НОВОСТНАЯ СТАТЬЯ	ПРОЦЕНТ СХОДСТВА; СТОЛБЧАТАЯ ДИАГРАММА СХОДСТВА
ДОКУМЕНТ	НОВОСТНОЕ СООБЩЕНИЕ ИЗ ИНТЕРНЕТ	ПРОЦЕНТ СХОДСТВА; СТОЛБЧАТАЯ ДИАГРАММА СХОДСТВА
ДОКУМЕНТ	ОНЛАЙНОВЫЙ КУРС ОБУЧЕНИЯ	ПРОЦЕНТ СХОДСТВА; СТОЛБЧАТАЯ ДИАГРАММА СХОДСТВА
ДОКУМЕНТ	ИСТОЧНИК ИНФОРМАЦИИ	ПРОЦЕНТ СХОДСТВА; СТОЛБЧАТАЯ ДИАГРАММА СХОДСТВА
ДОКУМЕНТ	МУЛЬТИМЕДИА	ПРОЦЕНТ СХОДСТВА; СТОЛБЧАТАЯ ДИАГРАММА СХОДСТВА
ДОКУМЕНТ	ПЕРСОНА	ВОЗМОЖНЫЙ АВТОР? (ИСТИННО/ЛОЖНО%); ЭКСПЕРТ? (ИСТИННО/ЛОЖНО%); ВОЗМОЖНО, ЗАИНТЕРЕСОВАН В (ИСТИННО/ЛОЖНО%); СВЯЗАННЫЕ ОБЪЕКТЫ, СОЗДАНИЕ АВТОРОМ
ДОКУМЕНТ	КЛИЕНТ	СВЯЗАННЫЕ СООБЩЕНИЯ ЭЛ.ПОЧТЫ, ПОСЛАНИЕ КЛИЕНТОМ; ВОЗМОЖНО, ЗАИНТЕРЕСОВАН В (ИСТИННО/ЛОЖНО%)

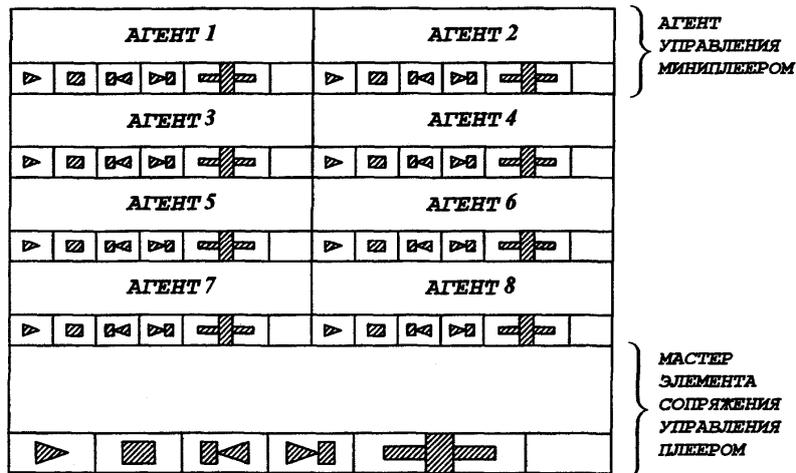
Фиг. 74

ТИП ЛУПЫ	ТИП ОБЪЕКТА	ИЗОБРАЖЕНИЕ
Эл. почта	Эл. почта	ПРОЦЕНТ СХОДСТВА; СТОЛБЧАТАЯ ДИАГРАММА СХОДСТВА
Эл. почта	СПИСОК РАСПРОСТРАНЕНИЯ ЭЛ. ПОЧТЫ	ЧЛЕНЫ, СОЗДАВШИЕ СООБЩЕНИЕ; ЧЛЕНЫ, КОТОРЫЕ МОГЛИ СОЗДАТЬ СООБЩЕНИЕ; ЧЛЕНЫ, КОТОРЫЕ ЯВЛЯЮТСЯ ЭКСПЕРТАМИ ПО СОДЕРЖАНИЮ СООБЩЕНИЯ
Эл. почта	ОБЩЕДОСТУПНАЯ ПАПКА ЭЛ. ПОЧТЫ	ЭТО ОБЩЕДОСТУПНАЯ ПАПКА? (ИСТИННО/ЛОЖНО?); СООБЩЕНИЯ ЭЛ. ПОЧТЫ В ОБЩЕДОСТУПНОЙ ПАПКЕ, КОТОРЫЕ ОТНОСЯТСЯ К ЭТОМУ ДОКУМЕНТУ; СООБЩЕНИЯ ЭЛ. ПОЧТЫ В ОБЩЕДОСТУПНОЙ ПАПКЕ, КОТОРЫЕ МОГУТ ОТНОСИТЬСЯ К ЭТОМУ ДОКУМЕНТУ
Эл. почта	СОБЫТИЕ	ТЕКУЩИЕ ДЕЛА
Эл. почта	НОВОСТНАЯ СТАТЬЯ	ТЕКУЩИЕ ДЕЛА
Эл. почта	НОВОСТНОЕ СООБЩЕНИЕ ИЗ ИНТЕРНЕТ	ТЕКУЩИЕ ДЕЛА
Эл. почта	ОНЛАЙНОВЫЙ КУРС ОБУЧЕНИЯ	ТЕКУЩИЕ ДЕЛА
Эл. почта	ИСТОЧНИК ИНФОРМАЦИИ	ТЕКУЩИЕ ДЕЛА
Эл. почта	МУЛЬТИМЕДИА	ТЕКУЩИЕ ДЕЛА
Эл. почта	ПЕРСОНА	ТЕКУЩИЕ ДЕЛА
Эл. почта	КЛИЕНТ	ТЕКУЩИЕ ДЕЛА

Фиг. 75



Фиг. 76



Фиг. 77

ИНСТРУМЕНТАЛЬНОЕ СРЕДСТВО	ОБЫЧНЫЙ WEB-БРАУЗЕР	ИНФОРМАЦИОННЫЙ АГЕНТ
Назад	ПЕРЕХОДИТ К ПРЕДЫДУЩЕЙ WEB-СТРАНИЦЕ	ПЕРЕХОДИТ К ПРЕДЫДУЩЕМУ АГЕНТУ
Вперед	ПЕРЕХОДИТ К СЛЕДУЮЩЕЙ WEB-СТРАНИЦЕ	ПЕРЕХОДИТ К СЛЕДУЮЩЕМУ АГЕНТУ
Домой	ПЕРЕХОДИТ К ИСХОДНОЙ СТРАНИЦЕ	ПЕРЕХОДИТ К ИСХОДНОМУ АГЕНТУ
Стоп	ОСТАНАВЛИВАЕТ ТЕКУЩУЮ ОПЕРАЦИЮ	ОСТАНАВЛИВАЕТ ТЕКУЩУЮ ОПЕРАЦИЮ (ИЛИ ТРЕБОВАНИЕ СЕМАНТИЧЕСКОГО ЗАПРОСА)
Обновить	ОБНОВЛЯЕТ ТЕКУЩУЮ WEB-СТРАНИЦУ	ОБНОВЛЯЕТ ТЕКУЩИЙ СЕМАНТИЧЕСКИЙ ЗАПРОС И ОБНОВЛЯЕТ СЕМАНТИЧЕСКУЮ СРЕДУ (И UI И ДЕРЕВООБНОЕ ПРЕДСТАВЛЕНИЕ, ОТРАЖАЮЩЕ ЭТО)
Печать	РАСПЕЧАТЫВАЕТ ТЕКУЩУЮ WEB-СТРАНИЦУ	РАСПЕЧАТЫВАЕТ ТЕКУЩИЕ ОТОБРАЖАЕМЫЕ ИНФОРМАЦИОННЫЕ ОБЪЕКТЫ (РЕЗУЛЬТАТЫ СЕМАНТИЧЕСКОГО ЗАПРОСА)

Фиг. 78

ИНСТРУМЕНТ. СРЕДСТВО	ПРОГРАММА АНАЛИЗА ФАЙЛОВОЙ СИСТЕМЫ ИЛИ ПРОГРАММА ПРОСМОТРА ДОКУМЕНТА	СЕМАНТИЧЕСКИЙ БРАУЗЕР
Создать	СОЗДАЕТ НОВЫЙ ФАЙЛ (ЧЕРЕЗ СВОЕ ПРИЛОЖЕНИЕ) ИЛИ ПАПКУ	СОЗДАЕТ НОВОГО ИНТЕЛЛЕКТУАЛЬНОГО АГЕНТА (ПОСРЕДСТВОМ МАСТЕРА «СОЗДАТЬ ИНТЕЛЛЕКТУАЛЬНОГО АГЕНТА») ИЛИ НОВЫЙ ЭЛЕМЕНТ СОПРЯЖЕНИЯ (ЧЕРЕЗ МАСТЕРА «СОЗДАТЬ ЭЛЕМЕНТ СОПРЯЖЕНИЯ»)
Открыть	ОТКРЫВАЕТ ВЫБРАННЫЙ ФАЙЛ (ЧЕРЕЗ СВОЕ ПРИЛОЖЕНИЕ) ИЛИ ПАПКУ	ОТКРЫВАЕТ АГЕНТА (ЧЕРЕЗ ДИАЛОГ «ОТКРЫТЬ АГЕНТА»), ДОКУМЕНТЫ ФАЙЛОВОЙ СИСТЕМЫ (ДЛЯ ИМПОРТИРОВАНИЯ ИХ В СЕМАНТИЧЕСКУЮ СРЕДУ И ПРИДАНИЯ ИМ ИНТЕЛЛЕКТУАЛЬНОСТИ И ЗАВИСИМОСТИ ОТ КОНТЕКСТА/ВРЕМЕНИ; ОСУЩЕСТВЛЯЕТ ПОИСК ДОКУМЕНТОВ ЛОКАЛЬНОЙ ФАЙЛОВОЙ СИСТЕМЫ, ОТКРЫВАЕТ URL (КАК ИНТЕРНЕТ-HTTP ИЛИ FTP-АДРЕС); ИЛИ ОТКРЫВАЕТ ТЕКУЩИЙ ОБЪЕКТ В БУФЕРЕ ОБМЕНА
Открыть диалог Документ	ОТКРЫВАЕТ ДИАЛОГОВОЕ ОКНО, ПОЗВОЛЯЮЩЕЕ ПОЛЬЗОВАТЕЛЮ ПРОСМАТРИВАТЬ СВОЮ СРЕДУ ФАЙЛОВОЙ СИСТЕМЫ ДЛЯ ОТКРЫТИЯ НУЖНОГО ДОКУМЕНТА ИЛИ ПРОСТО ДЛЯ ПРОСМОТРА СРЕДЫ	ОТКРЫВАЕТ ДИАЛОГОВОЕ ОКНО ПОЗВОЛЯЮЩЕЕ ПОЛЬЗОВАТЕЛЮ ПРОСМАТРИВАТЬ СВОЮ СРЕДУ ФАЙЛОВОЙ СИСТЕМЫ ДЛЯ ОТКРЫТИЯ НУЖНОГО ДОКУМЕНТА ИЛИ ПРОСТО ДЛЯ ПРОСМОТРА СРЕДЫ
Сохранить	СОХРАНЯЕТ ТЕКУЩИЙ ДОКУМЕНТ	СОХРАНЯЕТ ТЕКУЩЕГО АГЕНТА В СПИСКЕ «ПРЕДПОЧТЕНИЯ» (ЕСЛИ ОН УЖЕ НЕ БЫЛ ПРЕДПОЧТЕНИЕМ); ИЛИ КОПИРУЕТ ТЕКУЩЕГО АГЕНТА В ЭЛЕМЕНТ СОПРЯЖЕНИЯ

Фиг. 79А

Копировать	<p>КОПИРУЕТ ИНФОРМАЦИЮ ТЕКУЩЕГО ВЫБРАННОГО ФАЙЛА, ПАПКИ ИЛИ ТЕКСТА/ГРАФИКИ/СВЯЗИ В БУФЕР ОБМЕНА ДЛЯ ВСТАВКИ В РАЗЛИЧНЫЕ ПАПКИ (В СЛУЧАЕ ФАЙЛОВ И/ИЛИ ПАПКИ) ИЛИ ДЛЯ ВСТАВКИ В ТОТ ЖЕ ИЛИ НОВЫЙ ДОКУМЕНТ (В СЛУЧАЕ ИНФОРМАЦИИ ТЕКСТА/ГРАФИКИ/СВЯЗИ)</p>	<p>ИЗ ПАНЕЛИ ИНСТРУМЕНТОВ СЕМАНТИЧЕСКОГО БРАУЗЕРА, ЕСЛИ ИНТЕЛЛЕКТУАЛЬНЫЙ АГЕНТ ВЫБРАН ИЛИ ОТКРЫТ, SQML ДЛЯ АГЕНТА КОПИРУЕТСЯ В БУФЕР ОБМЕНА С ИСПОЛЬЗОВАНИЕМ СОБСТВЕННОГО SQML-ФОРМАТА БУФЕРА ОБМЕНА. ЭТО ЗАПУСКАЕТ ФУНКЦИОНАЛЬНЫЕ СРЕДСТВА ИНТЕЛЛЕКТУАЛЬНОЙ ЛУПЫ - ПОЛЬЗОВАТЕЛЬ МОЖЕТ ВЫБРАТЬ ЛЕВОЙ ИНФОРМАЦИОННЫЙ ОБЪЕКТ В ОКНЕ ПРЕЗЕНТАТОРА И ВЫБРАТЬ ПИКТОГРАММУ «ВСТАВИТЬ КАК ИНТЕЛЛЕКТУАЛЬНУЮ ЛУПУ». ОТКРЫВАЕТСЯ ПАНЕЛЬ ИНТЕЛЛЕКТУАЛЬНОЙ ЛУПЫ, КОТОРАЯ ОТОБРАЖАЕТ РЕЗУЛЬТАТЫ РЕЛЯЦИОННОГО ЗАПРОСА, ОСНОВАННОГО НА ОБЪЕДИНЕНИИ SQML БАЗОВОГО ОБЪЕКТА С SQML БУФЕРА ОБМЕНА (ПУТЕМ ДОБАВЛЕНИЯ SQML БАЗОВОГО ОБЪЕКТА КАК АРГУМЕНТА К SQML БУФЕРА ОБМЕНА С СООТВЕТСТВУЮЩИМ ПРЕДИКАТОМ (ПО УМОЛЧАНИЮ) И СВЯЗИ ИДЕНТИФИКАТОРОМ ИЛИ URL РЕСУРСА ОБЪЕКТА. ЭТО СОЗДАЕТ НОВЫЙ СЕМАНТИЧЕСКИЙ SQML ПУТЕМ РЕЛЯЦИОННОГО ОБЪЕДИНЕНИЯ SQML ДЛЯ «ЛУПЫ» И «ОБЪЕКТА» ДЛЯ СОЗДАНИЯ «ИЗОБРАЖЕНИЯ». ПРЕДПОЧТИТЕЛЬНО, ТОЛЬКО ИНТЕЛЛЕКТУАЛЬНЫЕ АГЕНТЫ МОГУТ КОПИРОВАТЬСЯ, ВСТАВЛЯТЬСЯ И ИСПОЛЬЗОВАТЬСЯ КАК ИНТЕЛЛЕКТУАЛЬНЫЕ ЛУПЫ. ТОЛЬКО ОБЪЕКТЫ, ВОЗВРАЩЕННЫЕ ПО ЗАПРОСАМ В ИНТЕЛЛЕКТУАЛЬНЫХ АГЕНТОВ, МОГУТ СОДЕРЖАТЬ ГЛУБИННУЮ ИНФОРМАЦИЮ. ВСЕ ОБЪЕКТЫ МОЖНО ПЕРЕТАСКИВАТЬ, ОСТАВЛЯТЬ, КОПИРОВАТЬ, ВСТАВЛЯТЬ; МОЖНО АВТОМАТИЧЕСКИ ГЕНЕРИРОВАТЬ ПАЛИТРЫ КОНТЕКСТА (ВКЛЮЧАЯ НЕИНТЕЛЛЕКТУАЛЬНЫХ АГЕНТОВ - НАПРИМЕР, ИЗ ТАКИХ АГЕНТОВ ОТНОСЯЩИХСЯ К ЛОКАЛЬНОЙ ФАЙЛОВОЙ СИСТЕМЕ). ТАК «НЕИНТЕЛЛЕКТУАЛЬНЫЕ» ОБЪЕКТЫ МОГУТ БЫТЬ СДЕЛАНЫ ИНТЕЛЛЕКТУАЛЬНЫМИ, КАК ТОЛЬКО ОНИ ОКАЗЫВАЮТСЯ В СЕМАНТИЧЕСКОЙ «ПЕСОЧНИЦЕ» ИЛИ СРЕДЕ. ЕСЛИ ОПЕРАЦИЯ «КОПИРОВАТЬ» АКТИВИЗИРОВАНА НА ОБЪЕКТЕ В ОКНЕ РЕЗУЛЬТАТОВ ПРЕЗЕНТАТОРА (КОМАНДОЙ «КОПИРОВАТЬ» ВО ВСПЛЫВАЮЩЕМ МЕНЮ), ЭТО КОПИРУЕТ SQML ОБЪЕКТА В БУФЕР ОБМЕНА. ПРЕЗЕНТАТОР ВЫВОДИТ SQML ОБЪЕКТА ИЗ XML (ИЛИ SRML) ОБЪЕКТА ДОБАВЛЕНИЕМ ССЫЛКИ НА ИДЕНТИФИКАТОР ИЛИ URL/ПУТЬ.</p>
------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Фиг. 79В

Вставить	<p>ВСТАВЛЯЕТ СОДЕРЖАНИЕ БУФЕРА ОБМЕНА В ВЫБРАННОЕ МЕСТО. ЕСЛИ ЭТО ПАПКА, ТО ВСТАВЛЯЕТСЯ СОДЕРЖАНИЕ ВЫБРАННОГО ФАЙЛА ИЛИ ПАПКИ. ЕСЛИ ВЫБРАННОЕ МЕСТО — ДОКУМЕНТ, ТО ВСТАВЛЯЕТСЯ ТЕКСТ, ГРАФИКА ИЛИ СВЯЗЬ ИЗ ТОГО, ЧТО ЕСТЬ В БУФЕРЕ ОБМЕНА</p>	<p>ИЗ ПАНЕЛИ ИНСТРУМЕНТОВ СЕМАНТИЧЕСКОГО БРАУЗЕРА, ЕСЛИ ВЫБРАН ИЛИ ОТКРЫТ ИНТЕЛЛЕКТУАЛЬНЫЙ АГЕНТ, SQML ИЗ БУФЕРА ОБМЕНА ВСТАВЛЯЕТСЯ В SQML ИНТЕЛЛЕКТУАЛЬНОГО АГЕНТА ДЛЯ СОЗДАНИЯ НОВОГО SQML-БУФЕРА ИЛИ ФАЙЛА, ВКЛЮЧАЮЩЕГО РЕЛЯЦИОННЫЙ СЕМАНТИЧЕСКИЙ ЗАПРОС С ОБЪЕКТОМ БУФЕРА ОБМЕНА КАК АРГУМЕНТОМ ФИЛЬТРА. ИНСТРУМЕНТ ИНСТРУМЕНТАЛЬНОЙ ПАНЕЛИ ЗАПУСКАЕТСЯ, ЕСЛИ БУФЕР ОБМЕНА СОДЕРЖИТ ТО, ЧТО МОЖЕТ ВСТАВЛЯТЬСЯ В АГЕНТА. СЕМАНТИЧЕСКОМУ БРАУЗЕРУ ПРОВЕРЯЕТ ЭТО ПУТЕМ ПРОВЕРКИ ФОРМАТА СОДЕРЖАНИЯ БУФЕРА. ЕСЛИ ЭТО SQML-ФОРМАТ ИЛИ МОЖЕТ БЫТЬ ПРЕОБРАЗОВАН В SQML, ТО СРЕДСТВО ЗАПУСКАЕТСЯ. ПРИ НЕОБХОДИМОСТИ, НЕЯВНОЕ ПРЕОБРАЗОВАНИЕ В SQML ВЫПОЛНЯЕТСЯ ПЕРЕД АКТИВИЗАЦИЕЙ ОПЕРАЦИИ ВСТАВКИ. ЕСЛИ SQML В БУФЕРЕ ОБМЕНА СООТВЕТСТВУЕТ ИНТЕЛЛЕКТУАЛЬНОМУ АГЕНТУ, СЕМАНТИЧЕСКОМУ БРАУЗЕРУ МОЖЕТ ПОДДЕРЖИВАТЬ ЕГО, ЕСЛИ АГЕНТ ПРИНАДЛЕЖИТ К ТОМУ ЖЕ АГЕНТСТВУ. В ЭТОМ СЛУЧАЕ ЗАПРОС ПРИВЕДЕТ К ПОДЗАПРОСУ, КОТОРЫЙ ФИЛЬТРУЕТ РЕЗУЛЬТАТЫ АГЕНТА ПОЛУЧАТЕЛЯ РЕЗУЛЬТАТАМИ АГЕНТА ИСТОЧНИКА. ЕСЛИ SQML В БУФЕРЕ ОБМЕНА ПРИНАДЛЕЖИТ К ТЕКУЩЕМУ ВЫБРАННОМУ АГЕНТУ (КОГДА ЖЕЛАТЕЛЬНО УКАЗАТЬ «КОПИРОВАТЬ» И ЗАТЕМ СРАЗУ «ВСТАВИТЬ» В ТОМ ЖЕ САМОМ ИНТЕЛЛЕКТУАЛЬНОМ АГЕНТЕ), ИНСТРУМЕНТАЛЬНОЕ СРЕДСТВО БУДЕТ ОТКЛЮЧЕНО, ТАК КАК ЭТО БУДЕТ ИЗЫМТОЧНОЙ ОПЕРАЦИЕЙ (ВВИДУ ТОГО ЧТО ПОЛЬЗОВАТЕЛЬ КОПИРУЕТ ЧАСТЬ ТЕКСТА В ДОКУМЕНТ И ЗАТЕМ ПОВЕРХ ВСТАВЛЯЕТ ТУ ЖЕ ЧАСТЬ ТЕКСТА). ЕСЛИ ОПЕРАЦИЯ «ВСТАВИТЬ» АКТИВИЗИРОВАНА НА ОБЪЕКТЕ В ОКНЕ РЕЗУЛЬТАТОВ ПРЕЗЕНТАТОРА (ЧЕРЕЗ ПИКТОГРАММУ «ВСТАВИТЬ КАК ИНТЕЛЛЕКТУАЛЬНУЮ ЛУПУ» НА ОБЪЕКТЕ), ТО ОТКРЫВАЕТСЯ ПАНЕЛЬ ИНТЕЛЛЕКТУАЛЬНОЙ ЛУПЫ, ОТОБРАЖАЮЩАЯ РЕЗУЛЬТАТЫ РЕЛЯЦИОННОГО ЗАПРОСА КАК ОБЪЕДИНЕНИЕ SQML БАЗОВОГО ОБЪЕКТА С SQML БУФЕРА ОБМЕНА (ПУТЕМ ДОБАВЛЕНИЯ SQML БАЗОВОГО ОБЪЕКТА КАК АРГУМЕНТА К SQML БУФЕРА ОБМЕНА С СООТВЕТСТВУЮЩИМ ПРЕДИКАТОМ (ПО УМОЛЧАНИЮ) И СВЯЗИ ИДЕНТИФИКАТОРОМ ИЛИ URL РЕСУРСА ОБЪЕКТА. ЭТО СОЗДАЕТ НОВЫЙ СЕМАНТИЧЕСКИЙ SQML ПУТЕМ РЕЛЯЦИОННОГО ОБЪЕДИНЕНИЯ SQML ДЛЯ «ЛУПЫ» И «ОБЪЕКТА» ДЛЯ СОЗДАНИЯ «ИЗОБРАЖЕНИЯ».</p>
----------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

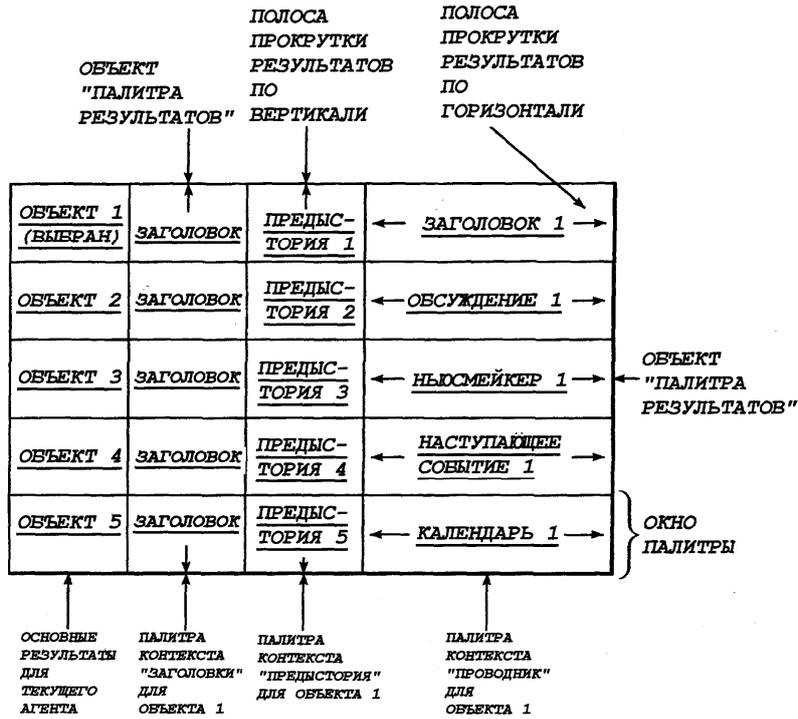
Фиг. 79С

Свойства	<p>ОТОБРАЖАЕТ ДИАЛОГОВОЕ ОКНО СВОЙСТВ ВЫБРАННОГО ФАЙЛА ИЛИ ПАПКИ, ИЛИ ТЕКУЩЕГО ДОКУМЕНТА.</p>	<p>ЕСЛИ ВЫБРАН ИЛИ ОТКРЫТ АГЕНТ (ИНТЕЛЛЕКТУАЛЬНЫЙ ИЛИ НЕИНТЕЛЛЕКТУАЛЬНЫЙ), ОТОБРАЖАЕТСЯ ДИАЛОГОВОЕ ОКНО, ПОКАЗЫВАЮЩЕЕ СВОЙСТВА АГЕНТА (ИМЯ, ОПИСАНИЕ, ДАТА/ВРЕМЯ СОЗДАНИЯ, ДАТА/ВРЕМЯ ПОСЛЕДНЕГО ИСПОЛЬЗОВАНИЯ И Т.Д.)</p>
Удалить	<p>УДАЛЯЕТ ТЕКУЩИЙ ВЫБРАННЫЙ ФАЙЛ, ПАПКУ, ГРАФИЧЕСКИЙ ОБЪЕКТ СВЯЗЬ ИЛИ ТЕКСТ</p>	<p>УДАЛЯЕТ ТЕКУЩИЙ ВЫБРАННЫЙ ИЛИ ОТКРЫТОГО АГЕНТА И ОБНОВЛЯЕТ СРЕДУ ДЛЯ ОТОБРАЖЕНИЯ НОВОГО СОСТОЯНИЯ.</p>

Фиг. 79D

<p>Перетасовать и Оставить</p>	<p>КОПИРУЕТ ИЛИ ПЕРЕМЕШАЕТ ВЫБРАННЫЙ ФАЙЛ, ПАПКУ, ГРАФИКУ ИЛИ СВЯЗЬ ОТ ИСТОЧНИКА К АДРЕСАТУ. ЕСЛИ ОБЪЕКТ-ФАЙЛ ИЛИ ПАПКА, СОДЕРЖАНИЕ КОПИРУЕТСЯ В ПАПКУ АДРЕСАТ. ЕСЛИ ОБЪЕКТ-ГРАФИКА ИЛИ ТЕКСТ, СОДЕРЖАНИЕ КОПИРУЕТСЯ В РАЗДЕЛ «ВСТАВИТЬ» В ТОМ ЖЕ ДОКУМЕНТЕ ИЛИ НОВОМ ДОКУМЕНТЕ.</p>	<p>СВЯЗЫВАЕТ ФАЙЛЫ, ПАПКИ, URL, ТЕКСТ ИЛИ ДРУГУЮ ЛОКАЛЬНУЮ КОНТЕКСТНУЮ СВЯЗЬ, КОТОРЫЕ ПЕРЕТАСКИВАЮТСЯ К ИНТЕЛЛЕКТУАЛЬНОМУ АГЕНТУ ДЛЯ СОЗДАНИЯ НОВОГО ИНТЕЛЛЕКТУАЛЬНОГО АГЕНТА, КОТОРЫЙ СОДЕРЖИТ СЕМАНТИЧЕСКИЙ ЗАПРОС, ОТРАЖАЮЩИЙ ВЗАИМОСВЯЗИ МЕЖДУ ПЕРЕТАСКИВАЕМЫМ ОБЪЕКТОМ (КОНТЕКСТОМ) И ИНТЕЛЛЕКТУАЛЬНЫМ АГЕНТОМ (АДРЕСАТОМ). РЕЗУЛЬТАТОМ ЯВЛЯЕТСЯ НОВЫЙ ИНТЕЛЛЕКТУАЛЬНЫЙ АГЕНТ С НОВЫМ SQML-БУФЕРОМ, КОТОРЫЙ ИМЕЕТ ИСХОДНЫЙ SQML ИНТЕЛЛЕКТУАЛЬНОГО АГЕНТА (АДРЕСАТА), НА КОТОРОГО ПЕРЕНЕСЕНЫ ИСТОЧНИКИ (ФАЙЛЫ, ПАПКИ, URL), ОТФИЛЬТРОВАННЫЕ ССЫЛКАМИ НА ИСТОЧНИК. НОВЫЙ SQML «КОМПИЛИРУЕТСЯ» КЛИЕНТОМ, ИНДИВИДУАЛЬНЫЕ SQML ПОСЫЛАЮТСЯ НА СЕРВЕР, ОПРЕДЕЛЕННЫЕ XML МЕТАДААННЫМИ ССЫЛОК НА ИСТОЧНИК. ЕСЛИ ИСТОЧНИК - ДОКУМЕНТ ФАЙЛА, МЕТАДААННЫЕ ДОКУМЕНТА ВЫДЕЛЯЮТСЯ И ПОСЫЛАЮТСЯ НА XMLWEB-СЕРВИС КАК SQML-АРГУМЕНТ. ЕСЛИ ИСТОЧНИК ИМЕЕТ МНОЖЕСТВО ФАЙЛОВ ИЛИ ЯВЛЯЕТСЯ ПАПКОЙ, ТО ПОСЫЛАЮТСЯ МЕТАДААННЫЕ ФАЙЛОВ В ПАПКЕ (ОПРЕДЕЛЯЕТСЯ СЛЕДУЕТ ЛИ ИСПОЛЬЗОВАТЬ АРГУМЕНТЫ ИСТОЧНИКА КАК ФИЛЬТР ОБЪЕДИНЕНИЯ ИЛИ ПЕРЕСЕЧЕНИЯ). ЕСЛИ ИСТОЧНИК - URL (HTTP, FTP), ТО URL ЗАГРУЖАЕТСЯ, И ЗАТЕМ МЕТАДААННЫЕ ДОКУМЕНТА, НА КОТОРЫЙ ССЫЛАЕТСЯ URL, ПОСЫЛАЮТСЯ КАК SQML-АРГУМЕНТ НА XMLWEB-СЕРВИС. БРАУЗЕР СНАЧАЛА ПОДТВЕРЖДАЕТ, ЧТО ИСТОЧНИК ССЫЛАЕТСЯ НА ТИП ОБЪЕКТА, КОТОРЫЙ СЕРВЕР ПОНИМАЕТ (ДОКУМЕНТ, ОБЪЕКТ ЭЛЕКТРОННОЙ ПОЧТЫ, ПЕРСОНА). ЕСЛИ ОБЪЕКТ НЕ ДЕЙСТВИТЕЛЕН, БРАУЗЕР ИГНОРИРУЕТ ЗАПРОС И/ИЛИ УВЕДОМЛЯЕТ ПОЛЬЗОВАТЕЛЯ СООБЩЕНИЕМ ДИАЛОГА ОПОВЕЩЕНИЙ.</p>
<p>Иерархический вид объекта</p>	<p>ОТОВАЖДАЕТ ИЕРАРХИЧЕСКИЙ ВИД ПАПКИ В СРЕДЕ ФАЙЛОВОЙ СИСТЕМЫ</p>	<p>ОТОВАЖДАЕТ ИЕРАРХИЧЕСКИЙ ВИД В СЕМАНТИЧЕСКОЙ СРЕДЕ, ВКЛЮЧАЯ ВИДЫ АГЕНТА, АГЕНТСТВА, АГЕНТОВ, ПРОСМАТРИВАЕМЫХ КАК «ВСЕ», «КОНТЕКСТ», «КАТЕГОРИЯ», «ВРЕМЯ», «АГЕНТСТВО», «ТИП ИНФОРМАЦИИ», «ЭЛЕМЕНТ СОПРЯЖЕНИЯ», И АГЕНТСТВА, ПРОСМАТРИВАЕМЫХ КАК «ВСЕ», «КОНТЕКСТ», «КАТЕГОРИЯ», «ВРЕМЯ», И Т.Д. (СМ. ВИДЫ ЭКРАНОВ).</p>

Фиг. 79Е



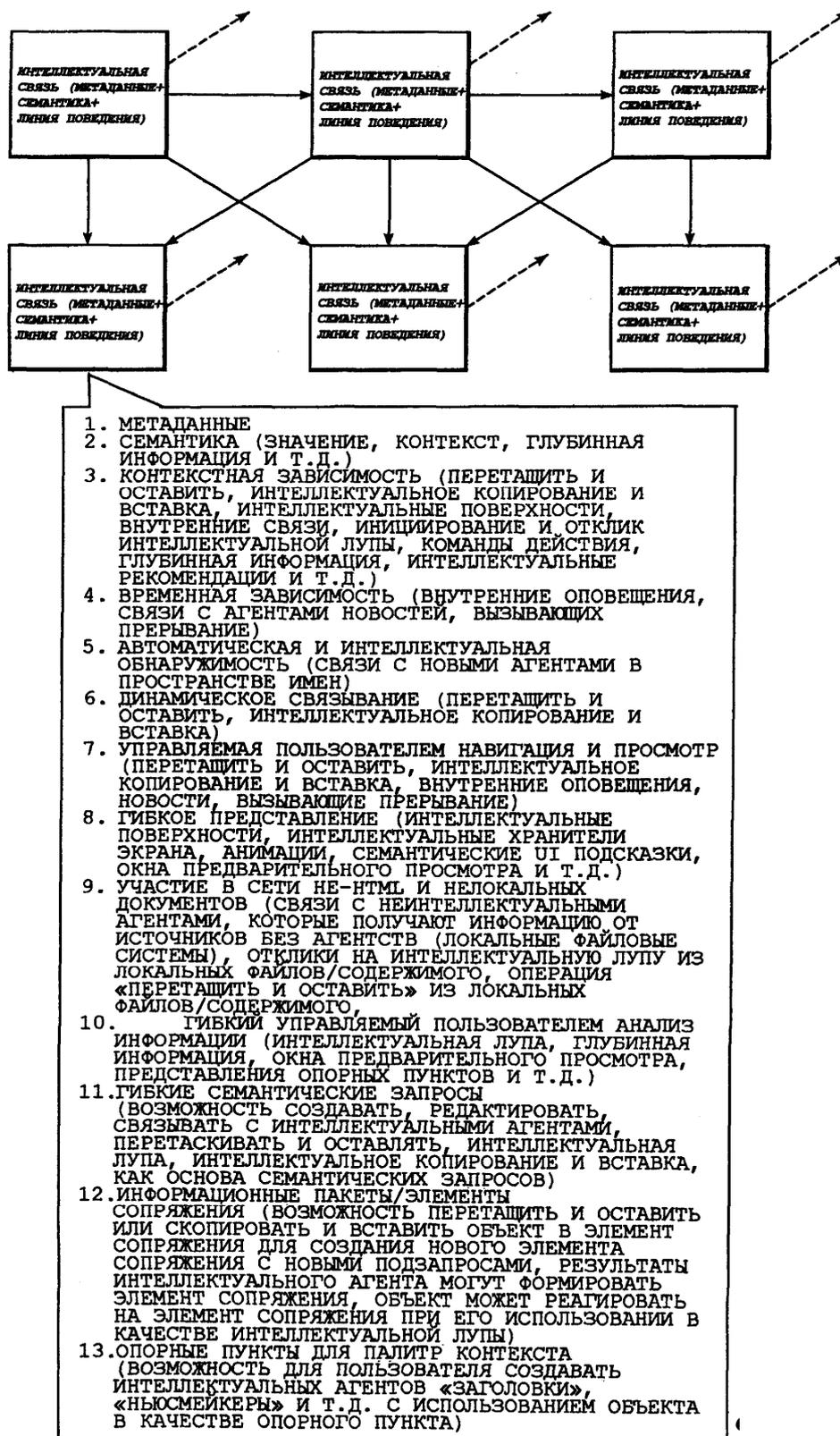
Фиг. 80

Последний элемент послан: 17 мая, 12:14 PM
[предварительный просмотр]
Следующее событие начнется: 28 июня 09:00 AM
[предварительный просмотр] <input type="button" value="↓"/>
Показать только элементы следующего типа: <input type="button" value="↓"/>
Показать только элементы, посланные последними:
Найдено всего 50 элементов [предварительный просмотр]

Фиг. 81

<p align="center">Уровни информационных инструментальных средств</p> <p>Специализированные – гипермедиа (интеллектуальные семантические связи)+метаданные+семантика/значение+»перетащить и оставить»+ интеллектуальное копирование и вставка + внутренние оповещения +интеллектуальная лупа +глубинная информация + внутренние команды действия+ внутренние связи + интеллектуальные агенты + шаблоны контекста (специальные Агенты) + элементы сопряжения + интеллектуальные рекомендации+ агенты новостей, вызывающих прерывание, + интеллектуальные, контекстно-зависимые связи с агентами новостей, вызывающих прерывание + семантические подсказки+ окна предварительного просмотра+ поверхности представления (незаметные, умеренные, экспрессивные, анимации и т.д.)+ интеллектуальные хранители экрана и т.д.</p>
<p align="center">Уровень представления</p> <p>Динамические, клиентские, конфигурируемые, контекстно-зависимые, время-зависимые «Поверхности» - клиентское преобразование XML в XHTML+TIME, SVG и т.д., с использованием XSLT</p>
<p align="center">уровень клиентских семантических запросов SQL (специализированные)</p>
<p align="center">уровень протокола (XML-Web-сервисы)</p>
<p align="center">Уровень серверных семантических запросов (SQL, XQUERY, XPATH и т.д.)</p>
<p align="center">Модель данных семантической сети (собственные данные, сохраненные в базе данных)</p>
<p align="center">Адаптеры источников семантических данных (DSAS) для семантических и несемантических источников данных Существующие хранилища для документов, электронной почты, контакты, события, клиентские данные, данные e-обучения и т.д.; хранилища метаданных (RDF, XML, базы данных и т.д.); специализированное отображение из источников данных на семантическую сеть</p>

Фиг. 82



Фиг. 83



Евразийская патентная организация, ЕАПВ

Россия, 109012, Москва, Малый Черкасский пер., 2/6