

УДК 681.51+004.94

ББК 32.988-5

ИНФОРМАЦИОННАЯ СИСТЕМА С ПОНЯТИЙНОЙ МОДЕЛЬЮ ПРЕДМЕТНОЙ ОБЛАСТИ

Выхованец В. С.¹

(ФГБУН Институт проблем управления
им. В.А. Трапезникова РАН, Москва)

Описывается интеллектуальная информационная система, основанная на понятийном моделировании предметной области. Понятийная модель состоит из понятийной структуры и описания содержания понятий. Понятийная структура определена как множество понятий, на которых заданы четыре отображения абстрагирования: обобщение, типизация, ассоциация и агрегация. Описание содержания понятий осуществляется с помощью таблиц базы данных. Существенным отличием используемой понятийной модели от других является описание ассоциации как обычного понятия. Показано, что благодаря семантической инвариантности области понятийной интерпретации удастся улучшить технологические и эксплуатационные характеристики информационной системы.

Ключевые слова: предметная область, информационная система, понятийная модель, понятийная структура, абстракции понятий, база знаний, репрезентация знаний.

1. Введение

Трудности управления современным крупным производством требуют внедрение информационных технологий, которые необходимы для ускоренного принятия решений на всех уровнях управления, адекватного реагирования на происходящие события, выработки и использования типовых рекоменда-

¹ Валерий Святославович Выхованец, доктор технических наук, доцент (valery@vykhovanets.ru).

ций и решений. Однако опыт внедрения современных информационных систем показывает, что реальная отдача от них оказывается значительно ниже, а сроки внедрения и связанные с ним затраты – существенно выше ожидаемых.

Последнее обусловлено несовершенством системы управления предприятием, неподходящей или отсталой инфраструктурой, недостаточной квалификацией персонала, размытой ответственностью, дублированием функций, слабым планированием и контролем, низкой корпоративной культурой [1]. Однако и современные информационные системы обладают рядом существенных недостатков, затрудняющих их эффективное использование по назначению.

Информационные системы строятся, как правило, по трехслойной архитектуре, которая включает в себя слой клиента, слой логики и слой базы данных. Основной особенностью таких систем является физическое разделение программ, отвечающих за хранение данных (слой базы данных), от программ, обрабатывающих данные (слой логики) и отображающих данные (слой клиента).

Усложнение решаемых задач привело к появлению информационных систем, имеющих четыре слоя: слой клиента, слой представления, слой логики и слой базы данных, где новый слой – слой представления – реализует информационную модель предметной области. Информационная модель предполагает использование некоторой формальной теории, позволяющей описывать объекты предметной области и связи между ними.

Все большую роль начинает приобретать концептуальное моделирование как направление в информационной технологии, с помощью которого строятся и встраиваются в информационные системы концептуальные модели различных предметных областей [14, 30]. Такие модели способствуют более быстрому построению и модификации приложений, решающих те или иные прикладные задачи.

Основная цель концептуального моделирования – формализация накопленных знаний о некоторой предметной области в форме, наиболее близкой пониманию пользователей и разработчиков информационной системы. С концептуальным подходом связывают ожидания эффективного описания сложных пред-

метных областей, повышения надежности и качества информационных систем, ускорения актуализации данных вслед за изменением в предметной области, обеспечения многократного использования моделей различных областей знания и т.п. [8]. Однако эти задачи еще не имеют законченной ни методологической, ни технологической проработки.

В последнее время активно развивается такое направление концептуального моделирования как онтологическое моделирование, где под онтологией понимается результат формализации некоторой области знаний с помощью концептуальной схемы. Концептуальная схема включает в себя единичные сущности предметной области и их классы (концепты), а также различного рода связи (отношения) между ними. Дополнительно к концептуальной схеме формулируются правила (аксиомы, теоремы, ограничения, функции интерпретации), принятые в предметной области [17].

Самыми распространенными типами отношений, используемых в онтологиях, является отношения категоризации (IS-A, KIND-OF, PART-OF, MEMBER-OF, INSTANCE-OF, CONTAINED-IN). Однако основную семантическую нагрузку в модели несет большое число специфических отношений между сущностями предметной области.

Правила онтологии задают утверждения, связывающие понятия и отношения между собой. Правила позволяют производить умозаключения и получать утверждения, которые не могут быть выражены посредством концептуальной схемы.

Для задания онтологий используются различные языки (OWL, CL, СусL и др.), различающиеся формальным аппаратом описания сущностей, концептов, связей и правил. Так, язык OWL использует дескрипционную логику¹, CL – *s*-выражения логики первого порядка, а СусL – исчисление предикатов.

Все перечисленные языки обладают одним существенным недостатком – высокой вычислительной сложностью вывода на знаниях. А языки, основанные на логике первого порядка и

¹ В OWL в качестве одного из расширений используется также язык, эквивалентный исчислению предикатов.

исчислении предикатов, к тому же еще и неразрешимы, что не гарантирует получения результата вывода за конечное время.

В свою очередь высокая вычислительная сложность языка дескрипционной логики связана с большим числом связей между концептами, вводимыми с помощью множества ролей (одноместных предикатов) для ограничения объемов концептов, а также с аксиомами вложенности концептов.

Настоящая статья посвящена описанию одной из реализаций четырехслойной корпоративной информационной системы, в основе которой лежит понятийное моделирование предметной области.¹

Суть подхода заключается в том, что понятийные модели предметной области строятся в формализме понятийной структуры, которая определена как множество понятий, на которых заданы только способы их образования в виде отображений типизации, агрегации, обобщения и ассоциации одних понятий в другие.²

В понятийной структуре, в отличие от других известных формальных теорий, таких как семантическая и ассоциативная сеть [18, 21], исчисление предикатов [19], модальная и дескрипционная логика [17, 20], концептуальный анализ [15], теория концептуальной зависимости [22], формальный анализ понятий [28], концептуальные графы [31], ER-модель [26] и др., ассоциация является обычным понятием, а не именованным видом связи, задающим в модели некоторую роль или отношение.

Язык понятийной модели, как и язык дескрипционной логики, является разрешимым, так как эквивалентен одноместному исчислению предикатов. Однако, в отличие от дескрипционной логики, язык понятийной модели не содержит средств описания отношений и ролей, так как все необходимые для

¹ *Пример реализации информационной системы, представленный в настоящей статье, публикуется с разрешения ЗАО «Инженерные системы и сервис» (Группа компаний «ЛАНИТ»).*

² *Под отображением понимается правило, по которому элементу одного множества ставится в соответствие элемент (или элементы) другого (или того же) множества. В то время как отношение – это подмножество декартова произведения множеств.*

вывода правила непосредственно содержатся в понятийной структуре.

Это позволяет повысить уровень абстракции модели и разработать информационную систему, для функционирования которой требуется небольшое число общих алгоритмов с небольшой вычислительной сложностью. Эти алгоритмы не зависят от предметной области, так как формулируются в предельно общих операциях над понятиями.

Благодаря этому удается улучшить вычислительные, технологические и эксплуатационные характеристики информационных систем с понятийными моделями предметной области.

2. Конкретные понятия

Наиболее систематичное исследование понятия содержится в работе [4], где понятие характеризуется как форма (вид) мысли и является «результатом обобщения предметов некоторого класса по определенной совокупности общих для предметов этого класса и – в совокупности отличительных для них признаков».

Полагается, что понятия образуются (определяются) при абстрагировании [10, 12], где абстрагирование рассматривается как один из основных процессов умственной деятельности человека, позволяющий под воздействием некоторой проблематики мысленно вычленить и превратить в самостоятельный объект рассмотрения отдельные свойства, стороны или состояния сущностей обозреваемой предметной области. При этом выделяются четыре вида понятий: единичные, простые, конкретные и абстрактные.

Единичные понятия. При образовании (определении) единичных понятий абстракция проявляется в способности мысленного выделения в предметной области уникальных сущностей и присвоения им имен. Иными словами, образование единичного понятия – это замена сущности некоторым знаком, тождественным сущности в некотором смысле.

Простые понятия. При образовании (определении) простых понятий абстракция проявляется как некоторое сознатель-

ное неведение, позволяющее сосредоточиться на одной стороне сущностей и игнорировать другие их стороны.

Простые понятия образуются путем объединения сущностей, подобных в некотором смысле. Простым понятиям присваивается уникальное имя, мыслимое как единичное понятие, и задается область допустимых проявлений (значений), мыслимая как множество единичных понятий.

Пример выражения простого понятия и принадлежащих ему сущностей – единичных понятий – приведен на рис. 1.

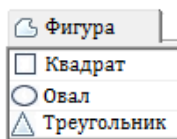


Рис. 1. Единичные и простые понятия

Конкретные понятия. Конкретные понятия образуются на основе объединения сущностей с одинаковыми признаками, позволяющими отличать сущности, принадлежащие понятию, от сущностей, понятию не принадлежащих. Выделяемые при этом признаки мыслятся как простые понятия, а имена понятий – как единичные понятия.

Определение 1. Конкретным понятием N называется упорядоченное множество из трех элементов: схемы $\text{shm } N$, интенционала $\text{int } N$ и экстенционала $\text{ext } N$:

$$(1) \quad N = \begin{cases} \text{shm } N = (P_0, P_1, \dots, P_{n-1}), \\ \text{int } N = \{(V_0^j, V_1^j, \dots, V_{n-1}^j) \mid j = \overline{0, m-1}\}, \\ \text{ext } N = \{E_0, E_1, \dots, E_{m-1}\}; \end{cases}$$

где N – имя понятия; $P_i, i = 0, 1, \dots, n-1$, – понятия-признаки; $V_0^j, V_1^j, \dots, V_{n-1}^j, j = 0, 1, \dots, m-1$, – наборы значений понятий-

признаков, составляющие интенционал понятия N ; $E_j, j = 0, 1, \dots, m - 1$, – сущности, принадлежащие понятию N .¹

Примеры выражения признаков конкретного понятия и принадлежащих ему сущностей приведены на рис. 2.

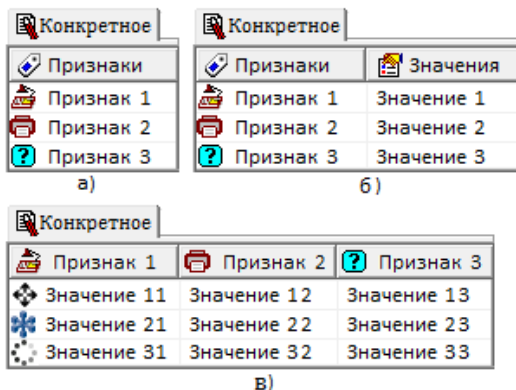


Рис. 2. Конкретное понятие: а) признаки; б) одна сущность; в) множество сущностей

Из определения (1) следует, что между сущностями E_j , составляющими экстенционал понятия N , и элементами $(V_0^j, V_1^j, \dots, V_{n-1}^j)$ его интенционала устанавливается взаимно однозначное соответствие. При этом сущности следует рассматривать как некоторые фрагменты предметной области, представляемые знаками, символами, образами и т.п. Для манипулирования сущностями в информационной системе их необходимо именовать и рассматривать как единичные понятия.

Следует заметить, что единичное понятие V (понятие-значение) и простое понятие P (понятие-признак) являются частными случаями конкретного понятия и в соответствии с (1) выражаются так:

¹ Здесь и далее множества элементов заключаются в фигурные скобки, а наборы элементов (элементы мультимножеств или упорядоченные множества с повторением элементов) – в круглые.

$$(2) \quad V = \begin{cases} \text{shm} V = (), \\ \text{int} V = \{V\}, \\ \text{ext} V = \{V\}; \end{cases} \quad P = \begin{cases} \text{shm} P = (P), \\ \text{int} P = \{V_j \mid j = \overline{0, m-1}\}, \\ \text{ext} P = \{V_0, V_1, \dots, V_{m-1}\}. \end{cases}$$

Таким образом, конкретное понятие представляется множеством сущностей, которые образуют его экстенционал или объем. Имя понятия – это знаковое выражение понятия, которому приписывается некоторый смысл. Схема или структура понятия задается набором признаков, характеризующих понятие. Интенционал или содержание понятия рассматривается как наборы значений взаимосвязанных признаков, позволяющие распознать сущности, принадлежащие понятию.

3. Абстрактные понятия

При образовании (определении) абстрактных понятий используются более сложные формы абстрагирования, основанные на установлении между понятиями отношений независимости, дифференциации и интеграции признаков (рис. 3).

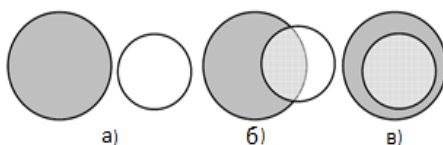


Рис. 3. Отношения признаков:

а) независимость; б) дифференциация; в) интеграция

Для образования абстрактных понятий применяются четыре абстракции: обобщение-специализация, типизация-конкретизация, агрегация-декомпозиция и ассоциация-индивидуализация [9, 11, 13, 25].

Обобщение и типизация, и обратные им абстракции – специализация и конкретизация, выражают общность понятий, проявляющуюся при дифференциации признаков. Ассоциация и агрегация, и обратные им – индивидуализация и декомпозиция, раскрывают структуру понятий, проявляющуюся при интеграции признаков.

Типизация-конкретизация. Типизация – порождение понятия на основе объединения экстенсионалов понятий с одинаковыми схемами.¹ При конкретизации понятия-типа выделяется одно из типизированных в нем понятий.

Для поиска сущностей в экстенсионале понятия-типа используется множество признаков, называемое ключом. Набор значений признаков, входящих в ключ, позволяет идентифицировать (отличать, именовать) сущности из экстенсионала понятия-типа.

Примеры выражения типизации-конкретизации приведены на рис. 4, где узел с именем «Ключ» используется для поиска и отображения в виде дочерних узлов сущностей понятия-типа путем ввода в наименование этого узла соответствующего ключа или его части. Если в ключ входит несколько признаков, то они разделяются знаком #.

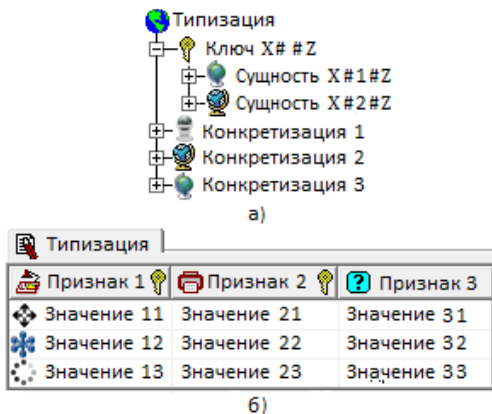


Рис. 4. Типизация-конкретизация:
а) абстрактных понятий; б) конкретных понятий

Обобщение-специализация. При обобщении происходит порождение нового понятия на основе нескольких подобных понятий, когда новое понятие сохраняет все или часть общих признаков исходных понятий и игнорирует другие признаки.

¹ В [25] типизация называется классификацией (англ. classification).

Обобщение – порождение нового понятия на основе пересечения схем обобщаемых понятий и объединения их экстенсионалов. При специализации, наоборот, из понятия-обобщения выделяется одно из обобщенных в нем понятий.

Как и при типизации-конкретизации для поиска сущностей в экстенсионале понятия-обобщения может использоваться ключ, который совпадает со схемой понятия-обобщения и может быть не уникальным.

Пример выражения обобщения-специализации приведен на рис. 5.

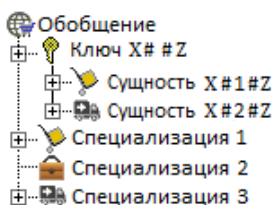


Рис. 5. Обобщение-специализация понятий

Следует заметить, что типизация-конкретизация является частным случаем обобщения-специализации. Однако при обобщении-специализации нельзя отобразить сущности понятия в виде таблицы (рис. 4б), так как обобщаемые понятия имеют разные схемы. В таблице возможно отображение только столбцов, принадлежащих схеме понятия-обобщения (принадлежащих общему набору признаков обобщаемых понятий).

Для указания на абстракцию обобщения-специализации связи между обобщаемыми понятиями проводят пунктирными линиями, для указания на абстракцию типизации-конкретизации – сплошными линиями (рис. 4, 5).

Ассоциация-индивидуализация. При ассоциации устанавливается взаимосвязь между сущностями одного и того же или разных понятий.¹ Ассоциация – порождение понятия на основе

¹ Ассоциация, традиционно рассматриваемая как роль, связь или отношение понятий, – это такой же вид мысли, как и мысль о части (декомпозиция), о целом (агрегация), о частном (специализация), об

объединения схем ассоциируемых понятий и ограниченного декартова произведения их экстенсионалов. Ассоциация выражает специфическое соединение сущностей ассоциируемых понятий, позволяющее переходить от сущностей одних ассоциированных понятий к сущностям других ассоциированных понятий. При индивидуализации из понятия-ассоциации выделяются ассоциированные в нем понятия.

Для поиска сущностей в экстенсионале понятия-ассоциации используется набор признаков, называемый связью и задающий ключ. Для характеристики связи между ассоциированными понятиями используется запись вида $L_0 : L_1 : \dots : L_{n-1}$, где n – число ассоциированных понятий, L_i – кардинальность вхождения i -го ассоциированного понятия, которая принимает одно из следующих значений: 1 (единичное вхождение), M (множественное вхождение).

Пример выражения ассоциации-индивидуализации приведен на рис. 6, где узел с именем «Ключ» используется для поиска и отображения в виде дочерних узлов сущностей понятия-ассоциации путем ввода в наименование этого узла признаков, входящих в связь (образующих ключ).

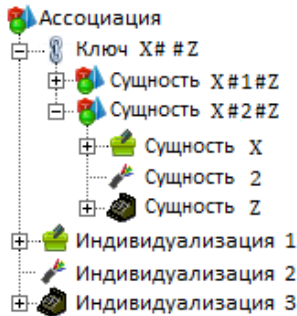


Рис. 6. Ассоциация-индивидуализация понятий

общем (обобщение), о классе (типизация), об экземпляре (конкретизация). Отсюда следует, что ассоциация может быть представлена как обычное понятие или как вид мысли, закономерно возникающей между двумя понятиями, связанными в каком-либо смысле.

Агрегация-декомпозиция. При агрегации понятие строится как соединение (композиция) других понятий. Агрегация – порождение понятия на основе объединения схем агрегируемых понятий и полного декартова произведения их экстенсионалов. При декомпозиции понятие-агрегат разделяется на входящие в него агрегированные понятия.

Пример выражения агрегации-декомпозиции приведен на рис. 7. В отличие от ассоциации, где между сущностями понятий устанавливаются только часть связей, в агрегации присутствуют все возможные связи. В этом случае в связь входит вся схема понятия-агрегата. Следует также заметить, что только понятие-агрегат может отображаться независимо по каждому агрегированному понятию (рис. 7б), для понятия-ассоциации это не имеет смысла.

Для указания на абстракцию ассоциации-индивидуализации связи между ассоциируемыми понятиями проводят пунктирными линиями, для указания на абстракцию агрегации-декомпозиции – сплошными линиями (рис. 6, 7).

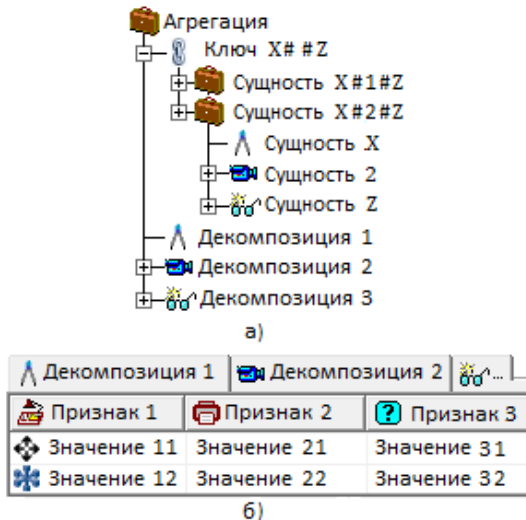


Рис. 7. Агрегация-декомпозиция:
а) абстрактных понятий; б) конкретных понятий

После описания абстракций понятий приведем их формальное определение [5].

Определение 2. Понятия N_j , $j = 0, 1, \dots, n - 1$, использованные для образования понятия N_G (N_T , N_C , N_A) путем обобщения (типизации, агрегации, ассоциации), называются обобщаемыми (типизируемыми, ассоциируемыми, агрегируемыми). При этом

$$(3) \quad \left\{ \begin{array}{l} \text{shm } N_G = \prod_{j=0}^{n-1} \text{shm } N_j, \\ \text{int } N_G \supseteq \bigcup_{j=0}^{n-1} \text{int } N_j, \\ \text{ext } N_G \supseteq \bigcup_{j=0}^{n-1} \text{ext } N_j; \end{array} \right. \quad \left\{ \begin{array}{l} \text{shm } N_T = \prod_{j=0}^{n-1} \text{shm } N_j, \\ \text{int } N_T = \bigcup_{j=0}^{n-1} \text{int } N_j, \\ \text{ext } N_T = \bigcup_{j=0}^{n-1} \text{ext } N_j; \end{array} \right.$$

$$(4) \quad \left\{ \begin{array}{l} \text{shm } N_A = \prod_{j=0}^{n-1} \text{shm } N_j, \\ \text{int } N_A \subseteq \times_{j=0}^{n-1} \text{int } N_j, \\ \text{ext } N_A \subseteq \times_{j=0}^{n-1} \text{ext } N_j; \end{array} \right. \quad \left\{ \begin{array}{l} \text{shm } N_C = \prod_{j=0}^{n-1} \text{shm } N_j, \\ \text{int } N_C = \times_{j=0}^{n-1} \text{int } N_j, \\ \text{ext } N_C = \times_{j=0}^{n-1} \text{ext } N_j; \end{array} \right.$$

где $\text{shm}(\text{int}, \text{ext})$ – схема (интенционал, экстенционал) понятия, $\Pi(\cup)$ – размеченное пересечение (объединение), выполняемое с повторением элементов, $\supseteq (\cup, \times)$ – включение (неразмеченное объединение, декартово произведение) множеств.

После определения абстракций понятий можно сделать следующие выводы:

– единичные понятия (понятия-значения) имеют пустую схему и именуются своим значением, а схема простого понятия (понятия-признака) состоит из имени этого понятия;

– сущности простого понятия (значения понятия-признака) являются единичными элементарными понятиями, а сами простые понятия (понятия-признаки) – объемными элементарными понятиями;

– простые понятия (понятия-признаки) образуются на основе абстракции обобщения, а конкретные понятия – на основе абстракции ассоциации.

4. Понятийная структура

Будем предполагать, что образование новых или выявление уже существующих понятий происходит в процессе изучения предметной области. При этом под углом зрения некоторой проблематики выделяются сущности, которые уже имеют или которым приписываются некоторые имена. Далее множество выявленных сущностей подвергается анализу на предмет установления их сходства и различия. Сходные сущности группируются, в результате чего происходит образование понятий, или наполнение уже имеющихся понятий проблемным содержанием.

В отличие от известных формализмов, где на понятиях задается множество отношений различной природы, будем использовать другой формализм – понятийную структуру, которую определим множеством понятий с четырьмя видами отображений, единственное назначение которых – показать способы образования понятий, способы их абстрагирования.¹

Определение 3. Понятийной структурой $S = (N, T, G, C, A)$ называется конечное множество понятий N , на которых заданы четыре конечных множества отображений вида $N \rightarrow N$: типизации T , обобщения G , агрегации C и ассоциации A .

Понятийная структура призвана в формализованном виде отразить результаты понятийного анализа предметной области и выражает отображения одних понятий в другие. Используемые при этом абстракции рассматриваются как умственные операции, необходимые и достаточные для мысленного выделения и превращения в отдельные понятия тех представлений, которые накоплены относительно формализуемой предметной области.

¹ Известные концептуальные модели обладают некоторой «концептуальной» незавершенностью. Они задаются множеством понятий и связей между ними, которые определяют смысловое содержание предметной области [24]. Однако любая такая связь также является понятием, с которым можно выполнять те же операции. Но в концептуальных моделях такие операции не предусмотрены.

Пример понятийной структуры приведен на рис. 8. Понятие «Тип проекта» определено как результат типизации единичных понятий «Строительство», «Реконструкция» «Модернизация».

Понятие «Учетный проект» есть результат ассоциации простых понятий «Наименование» и «Дата», а также понятия-типа «Тип проекта».

В свою очередь понятие «Руководитель проекта» есть результат обобщения таких понятий как «Руководитель отдела», «Главный инженер» и «Директор департамента».

И, наконец, понятие «Назначение» определено как ассоциация понятий «Учетный проект» и «Руководитель проекта».

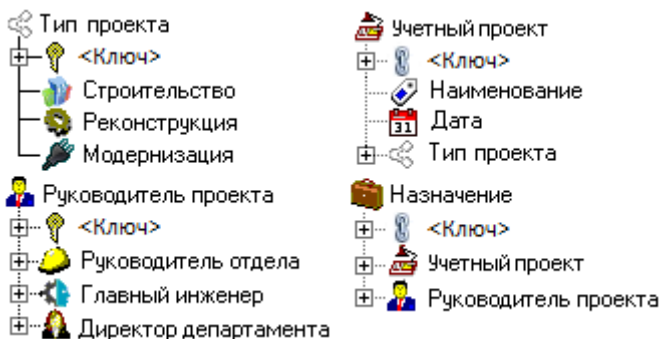


Рис. 8. Фрагмент понятийной структуры

Для поиска сущностей в экстенционалах дифференциальных понятий используется специальный узел с пиктограммой «Ключ», а для поиска сущностей в экстенционалах интегральных понятий – специальный узел с пиктограммой «Связь».

Схема понятия. Из определения понятия следует, что каждое понятие обладает схемой – набором признаков или простых понятий, на которых это понятие определено. Однако при задании понятийной структуры определению подлежат только отображения одних понятий в другие. Возникает задача определения схем понятий.

Для любого понятия из понятийной структуры S может быть найдена его схема по следующей рекуррентной процедуре:

- схема единичного понятия пуста;

- схема простого понятия состоит из имени этого понятия;
- схема понятия-дифференциации равна размеченному пересечению схем дифференцируемых понятий;
- схема понятий-интеграции равна размеченному объединению схем интегрируемых понятий;
- схема понятия, полученного в результате интеграции и дифференциации, равна размеченному объединению схем интегрируемых понятий, принадлежащая размеченному пересечению схем дифференцируемых понятий.¹

Следует обратить внимание на то, что возможность определения схем абстрактных понятий позволяет отображать последние как конкретные понятия в виде списка сущностей, раскрываемых через первичные понятия-признаки предметной области.

Для верификации понятийной структуры требуется проверить вычислимость схем всех входящих в нее понятий.² Требование вычислимости схем понятий является распространением свойства регулярности (фундированности) множеств на понятия. В этом случае вычислимость схем понятий гарантирует отсутствие определений понятий через самих себя, что в любой формальной или содержательной теории, претендующей на адекватность, признается недопустимым [7].

Таким образом, в результате понятийного анализа предметной области может быть получено ее онтологическое описание, заданное понятийной структурой. Принципиальными отличиями понятийной структуры от других концептуальных схем является:

- отсутствие разделения терминов на значения, признаки, понятия, связи и роли, а использование одного общего термина – понятия, частными проявлениями которого являются значения, признаки, понятия, связи и роли;

¹ *Необходимость одновременного использования абстракций дифференциации и интеграции при определении понятий связана с тем, что на одном и том же наборе понятий может быть определено несколько обобщений путем различного ограничения набора обобщающих признаков.*

² *Задача определения вычислимости схем понятий является алгоритмически разрешимой из-за конечности используемых множеств.*

- возможность представления ассоциаций как самостоятельных понятий, что позволяет, например, выразить обобщение ассоциативных связей;
- определение понятий, которые одновременно могут быть как обобщением, так и ассоциацией других понятий;
- семантическая инвариантность описания, не требующая для своей интерпретации привлечения предметных знаний.

5. Понятийная модель

В понятийной структуре предметной области задаются понятия и способы их абстрагирования. По понятийной структуре может быть также вычислена схема понятия – набор простых понятий (понятий-признаков), позволяющих рассматривать абстрактные понятия как конкретные. Однако какие наборы значений простых понятий определяют ту или иную сущность конкретных и абстрактных понятий, остается неясным. Для этого предназначен интенционал понятия, задающий правила соотнесения наборов значений простых понятий с той или иной сущностью из экстенционала описываемого понятия.

Определение 4. Понятийной моделью M предметной области называется ее понятийная структура S , дополненная описанием интенционалов D всех входящих в нее понятий, $M = (S, D)$.

Из определения 1 следует, что универсальной формой задания интенционала является перечисление наборов признаков каждой сущности, принадлежащей понятию. Использование этой формы на практике приводит к трудностям, если объем понятия достаточно велик. Другая форма описания интенционалов – использование процедур (формул, функций), разрешающих экстенционал определяемого понятия.¹

¹ Проблема представления простых понятий восходит к известной проблеме задания множеств. Множество называется разрешимым, если существует алгоритм, позволяющий определить, принадлежит ли элемент этому множеству или нет. Множество называется перечислимым, если существует алгоритм, порождающий это множество, т.е. последовательно выдающий все его элементы. Известно, что всякое разрешимое множество перечислимо. Однако обрат-

При использовании баз данных низкоуровневое представление понятийной модели предметной области может решаться только средствами, встроенными в систему управления базами данных [27]. Так единичные понятия задаются значениями простых типов данных: целыми числами, числами с плавающей запятой, датами, символами, строками с фиксированной и переменной длиной, двоичными данными.

Для задания простых понятий (понятий-признаков) могут использоваться ограничения целостности, сужающие области значений простых типов данных, или специальные таблицы, содержащие допустимые значения простого понятия. Конкретные понятия представимы таблицами, столбцы которых соответствуют простым понятиям из его схемы.

Сложнее обстоит дело с абстрактными понятиями. Представление абстрактного понятия-типа особых сложностей не вызывает и осуществляется с помощью запроса к базе данных, обрабатывающего записи одновременно из нескольких таблиц с одинаковым набором полей. Однако реализация абстрактного понятия-обобщения уже требует создания отдельной таблицы, где хранятся интенционалы сущностей, отсутствующих в экстенционалах обобщаемых понятий. Отсюда следует, что представление абстрактного обобщения осуществляется с помощью запроса к базе данных, в котором обработке подлежат все записи из таблицы понятия-обобщения и уникальные записи из таблиц обобщаемых понятий с набором полей, принадлежащих схеме понятия-обобщения.

Агрегация понятий представляется запросом, выполняющим произведение двух и более таблиц. Однако реализация абстрактного понятия-ассоциации уже не может быть сведена к запросу к базе данных и требует выделения отдельной таблицы, связывающей записи из двух и более таблиц ассоциируемых понятий.¹

ное неверно, поскольку существуют перечислимые, но неразрешимые множества [2].

¹ *Следует заметить, что в реализованной информационной системе для представления понятий-агрегатов также используются таблицы. Это связано с тем, что на практике оказывается востребованным*

Как показано ранее, для отображения интенционалов простых понятий используется табличная форма (рис. 9а). Табличные формы также могут быть сформированы для отображения интенционалов конкретных и абстрактных понятий. В этом случае таблица создается на основе схемы раскрываемого понятия (рис. 9б).

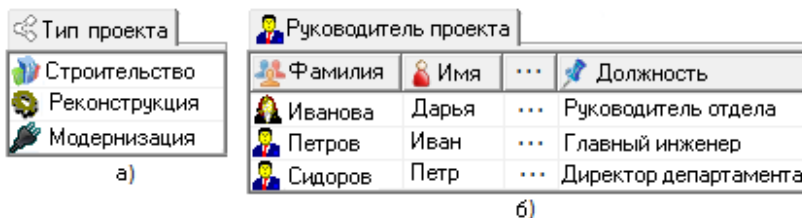


Рис. 9. Отображение интенционалов понятий:
 а) простых; б) конкретных и абстрактных

Здесь важно заметить, что табличная форма (список) на рис. 9б в целом выражает некоторое понятие, все ее строки – интенционал понятия, одна строка – конкретную сущность, а заголовки столбцов – схему понятия.

Таким образом, при использовании систем управления базами данных основным средством описания интенционалов понятий является перечисление наборов взаимосвязанных признаков в виде физической или виртуальной таблицы. Физическая таблица непосредственно хранится в памяти, а виртуальная таблица формируется динамически в результате выполнения некоторого запроса к базе данных. В этом случае строка (запись) таблицы соответствует описываемой сущности из экстенционала понятия, а столбец (поле) – признаку понятия.

небольшое число экземпляров понятия-агрегата. Для унификации процедур обработки понятий удобно такие экземпляры хранить в отдельной таблице.

6. Язык понятийной модели

Учитывая множество отображений абстрагирования, которые задаются между понятиями в понятийной структуре, для ее эффективного описания разработан специальный язык, формальная грамматика которого приведена на рис. 10.

Грамматика задана с точностью до пробельных знаков, которые могут появляться между терминальными и нетерминальными символами. Определяемые нетерминальные символы записаны в левой части правил вывода до знака \rightarrow , а терминальные символы заключены в апострофы. Альтернативы в правой части правил вывода перечислены через разделитель |, а необязательные вхождения символов грамматики заключены в квадратные скобки.

- 1) Модель \rightarrow Понятие [Модель]
- 2) Понятие \rightarrow Значение | Свойство | Признак
| Абстракция | Ссылка
- 3) Значение \rightarrow Строка
- 4) Свойство \rightarrow '=' Строка '#'
- 5) Признак \rightarrow Имя '=' Значение '#'
- 6) Абстракция \rightarrow Имя '=' '#' [Модель] '#'
- 7) Ссылка \rightarrow '=' '#' [Путь] '#'
- 8) Путь \rightarrow Имя '#' [Путь]
- 9) Имя \rightarrow Строка
- 10) Строка \rightarrow Символ [Строка]
- 11) Символ \rightarrow '\ ' | '\#' | '\=' | '\\' | Буква | Цифра | Знак

Рис. 10. Формальная грамматика языка модели

Модель предметной области состоит из непустого перечисления ее понятий (правило 1). Понятие может выражаться значением или свойством, быть признаком или абстракцией, а также являться ссылкой (правило 2).

Значение выражается произвольной строкой (правило 3), а свойство – терминальным символом «равно», после которого следует произвольная строка (правило 4). Свойства используются для уточнения описания понятий, невыражаемого на языке

понятийной модели. В этом случае интерпретация значений свойств осуществляется принимающей моделью программы.

Признак имеет имя, с которым связывается одно или несколько значений (правила 5 и 1). Абстракция также именуется, но с ней связывается некоторая подмодель (правило 6). Ссылка задается путем от корня модели к ранее описанному понятию (правила 7 и 8).

Имя, как и значение, выражается произвольной строкой (правило 9). Строка определена как произвольная последовательность символов (правило 10): букв, цифр и других знаков, допустимых в используемой кодировке (правило 11). Для обеспечения возможности использования в строке символов разметки (пробельные знаки, #, =, \) последние имеют специальное выражение (правило 11).

Ниже в качестве примера приведено описание понятийной структуры, показанной на рис. 8, где используются необходимые, но не показанные на рисунке, простые понятия «Строка» и «Дата».

```
Тип проекта = #
    = Типизация #
    Строительство #
    Модернизация #
    Реконструкция #
#
Учетный проект =#
    = Ассоциация #
    Наименование =#1
        =# Строка ##
#
    =# Дата ##
    =# Тип проекта ##
#
Руководитель проекта = #
```

¹ В случае необходимости на языке понятийной модели может быть выражена эквивалентность понятий. В примере понятие «Наименование» описано как эквивалентное понятию «Строка».

= Обобщение #
=# Руководитель отдела ##
=# Главный инженер ##
=# Директор департамента ##

Назначение =#
= Ассоциация #
=# Учетный проект ##
=# Руководитель проекта ##
#

7. Задачи предметной области

Использование понятийной структуры, задаваемой в виде дерева (рис. 8), и отображение сущностей из экстенционалов понятий в виде списков (рис. 9) не являются достаточными с точки зрения функционирования информационной системы, призванной автоматизировать решения заданного класса прикладных задач.

Абстрагируясь от конкретного содержания действий и процедур, составляющих алгоритмы решения той или иной прикладной задачи, можно сделать вывод о том, что все такие действия сводимы к трем абстрактным операциям над понятиями, а именно: созданию понятия, изменению понятия и удалению понятия. Таким образом, семантически инвариантной формой описания решений прикладных задач в моделируемой предметной области является описание, состоящее всего из трех элементарных операций.

Создание понятий. Операция создания понятия возникает при усложнении понятийной модели предметной области и состоит в определении имени нового понятия и задания способа его абстрагирования. После создания нового понятия автоматически вычисляется его схема. В зависимости от вида понятия в базе данных создается или таблица – для простых и конкретных понятий, а также абстрактных понятий-обобщений и понятий-ассоциаций, или представление (запрос) – для абстрактных

понятий-типизаций и понятий-агрегаций, способные перечислить сущности, принадлежащие созданному понятию.¹

Удаление понятий. Операция удаления понятия возникает в случае изменения представлений о предметной области и состоит в изменении описания всех понятий, в определения которых оно входит.

На рис. 11 показан пример отображения понятий «Объект» и «Проект» и доступных для них операций, активируемых через контекстное меню клиентского приложения.

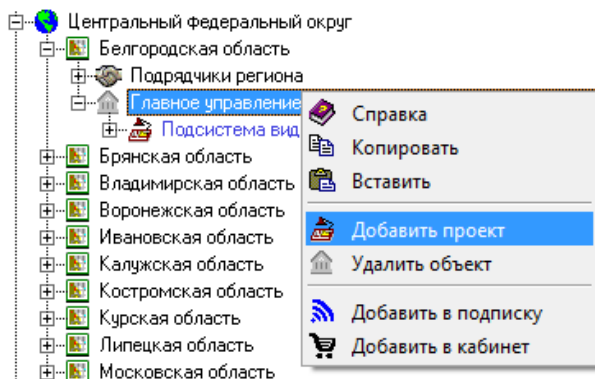


Рис. 11. Операции создания и удаления понятий

Изменение понятий. Операция изменения понятий используется при необходимости наполнить понятия конкретным предметным содержанием. Операции изменения выполняются над конкретными и простыми понятиями. В этом случае возможны три действия: редактирование существующей сущности (записи), удаление существующей сущности (записи) и добавление новой сущности (записи). Такие же действия выполняют-

¹ Следует заметить, что современные системы управления базами данных помимо табличного хранения данных могут выполнять хранимые процедуры [27], позволяющие формировать таблицы «на лету» путем вычисления как схем, так и содержания воспроизводимых ими понятий.

ся и над абстрактными понятиями-обобщениями и понятиями-ассоциациями, имеющими свои таблицы.

На рис. 12 показан пример отображения экстенционала понятия «Комплект» и его операций изменения.

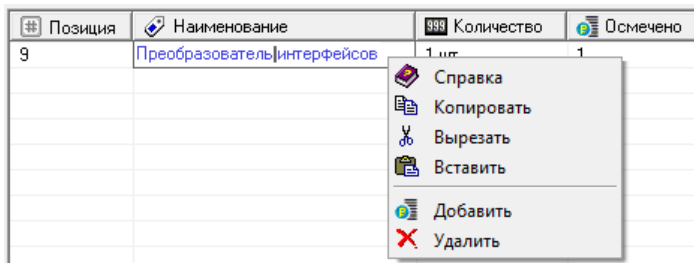


Рис. 12. Операции изменения понятия

Операции добавления и удаления сущностей понятия отображаются в контекстное меню приложения (рис. 11, 12), а операция редактирования сущности выполняется путем непосредственного изменения значения простого понятия или путем выбора из ниспадающего списка одного из его допустимых значений.

Для реализации операций создания, удаления и изменения понятий в системе управления базой данных могут быть предусмотрены специальные хранимые процедуры, обеспечивающие целостность и согласованность понятийной модели предметной области.

8. Информационная система

Многослойная структура информационной системы с понятийной моделью предметной области приведена на рис. 13.

Слой клиента реализован в виде клиентского приложения (рис. 14), функционирующего по принципу обозревателя (проводника, браузера): приложение получает данные от слоя знаний для отображения понятийной структуры в виде дерева (а), а интенционалов и экстенционалов понятий – в виде вкладок (б) и списка (в).



Рис. 13. Слои информационной системы

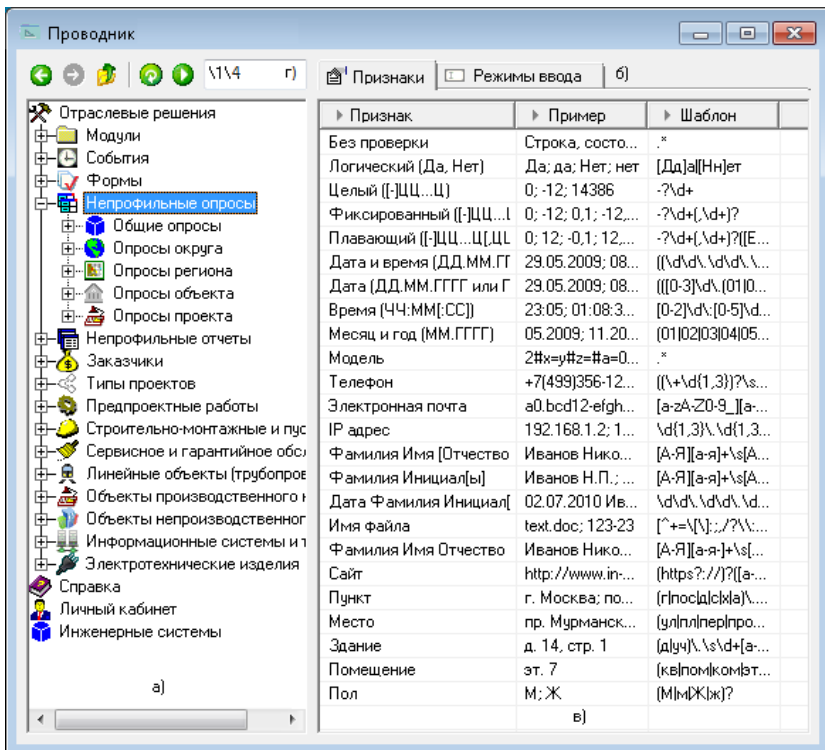


Рис. 14. Клиентское приложение:
а) дерево; б) вкладки; в) список; г) адресная строка

Для ускорения навигации по элементам формы используется кнопочное меню и адресная строка (г). Адресная строка также используется для поиска понятий по их именам.¹ Предусмотрена также возможность множественного и частичного отображение основной формы приложения, в которой отсутствует любой из перечисленных элементов.

Слой знаний служит для формирования данных в форматах, используемых клиентским приложением для отображения и манипулирования понятийной моделью.² Обмен данными осуществляется по двум интерфейсам: INode и IList. Интерфейс INode используется для отображения узла дерева и предоставляет его имя, пиктограмму, описание контекстного меню и вкладок, а также интерпретируемые тексты программ для изменения имени узла, обработки интерфейсных событий, отображения дочерних узлов. Интерфейс IList используется для отображения списка и предоставляет описание заголовка списка (столбцов), множество строк, состоящих из ячеек по числу столбцов, а также интерпретируемые тексты программ для изменения содержимого ячеек и обработки интерфейсных событий.

Для повышения эффективности информационной системы и сокращения времени реализации новых процедур предусмотрено использование унифицированным образом классов конкретных понятий (рис. 14), например:

- класса понятий, представляемых типовыми опросными формами;
- класса понятий, выражающих всевозможные отчеты, формируемые для анализа текущего состояния моделируемой предметной области;
- других классов понятий, имеющих типовое представление и однотипную обработку.

¹ В связи с использованием длинных имен понятий возникают значительные неудобства при указании пути понятия в понятийной структуре. По этой причине в адресной строке приложения отображается иерархический адрес текущего понятия, состоящий из последовательности номеров понятий в текущем пути.

² Почему второй слой информационной системы назван слоем знаний, будет показано далее.

Следует обратить внимание на то, что сама информационная система тоже имеет некоторую понятийную модель, работа с которой также происходит через клиентское приложение (рис. 14). В эту модель могут входить такие понятия как:

– модуль, подгружаемый в процессе работы клиентского приложения и служащий для реализации специфической функции отображения понятийной модели или решения специфической задачи предметной области;

– событие, регистрируемое в информационной системе и позволяющее задать обработчик для операций создания, удаления или изменения понятий;

– форма, создаваемая для реализации различных сценариев ввода и обработки данных пользователем;

– другие понятия, необходимые для реализации требований к модели конкретной предметной области.

Слой логики отвечает за выполнение операций над понятиями. Для реализации этого слоя могут использоваться как процедуры, запускаемые на выделенном для этого сервере (слой логики), так и хранимые процедуры системы управления базой данных (слой данных).

Для ограничения операций над понятиями, а также для формирования индивидуальных понятийных моделей, информационная система в слое логики реализует развитый механизм определения и наследования прав. Для любого понятия модели и группы пользователей (пользователя) могут быть заданы четыре вида прав: доступа, допуска, действия и владения (рис. 15).

Слой данных представлен системой управления базами данных. В случае развертывания больших информационных систем каждый слой может быть реализован не на одном, а на группе серверов, и содержать средства динамического распределения нагрузки на серверы следующего слоя. В этом случае слой данных реализуется в виде распределенной базы данных.

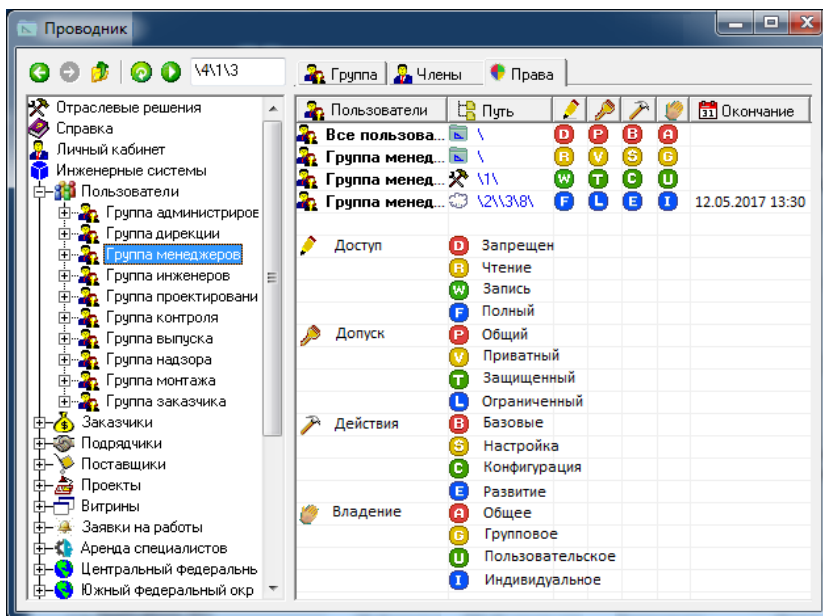


Рис. 15. Права

9. База знаний

В основе любой информационной системы, предназначенной для обработки знаний, лежит формальный аппарат представления знаний и манипулирования ими с целью имитации рассуждений человека для решения стоящих перед ним прикладных задач. В свою очередь под базой знаний понимается база данных, содержащая факты о некоторой предметной области, а также правила вывода, позволяющие автоматически выполнять умозаключения и получать новые утверждения об имеющихся или вновь вводимых фактах [3].

Информационная система с понятийной моделью предметной области $M = (S, D)$ может рассматриваться как база знаний. В этом случае понятийная структура S (понятия и способы их абстрагирования) задает правила вывода на знаниях, а интенсификации понятий D – факты (суждения) о предметной области.

Суждения. Фактами (суждениями) являются высказываниями о принадлежности сущностей предметной области экстенционалам понятий. В соответствии с формулой (1) сущность E принадлежит экстенсионалу понятия N , если и только если набор значений признаков сущности E , упорядоченный в соответствии со схемой понятия $\text{shm } N$, принадлежит интенсионалу понятия $\text{int } N$:

$$(5) \left(\prod_{(\forall P_i \in \text{shm } N)} P_i[E] \right) \in \text{int } N \leftrightarrow N(E),$$

где \prod – размеченное (упорядоченное) объединение, выполняемое с повторением элементов; $P_i[E]$ – функтор, возвращающий значение понятия-признака P_i сущности E ; \leftrightarrow – логическая связка двухстороннего следования; $N(E)$ – одноместный предикат принадлежности сущности E экстенсионалу понятия N , $N(E) \leftrightarrow E \in \text{ext } N$.

Согласно формуле (5) выполнимость одноместного предиката $N(E)$ определяется по текущему состоянию интенсиналов D , т.е. информационная система реализует модель открытого мира.¹

Умозаключения. Любое умозаключение может быть определено как переход от одного или нескольких суждений, составляющих посылку умозаключения, к утверждению – следствию умозаключения. Правила построения умозаключений задаются правилами вывода, принимаемыми в предметной области в качестве общезначимых, т.е. порождающих истинные утверждения при всех возможных посылках.

Правила построения умозаключений в рассматриваемом случае задаются правилами вывода, формализованными в понятийной структуре предметной области, а сама понятийная структура рассматривается как формальная теория, которая сохраняет истинность всех выводимых в ней следствий.

¹ В модели открытого мира нарушается монотонность вывода, заключающаяся в том, что если в результате вывода получено некоторое утверждение, то при поступлении новых фактов выводимость этого утверждения не должна исчезнуть.

Из формул (3) и (4) получаем логические высказывания, задающие правила вывода на знаниях:

$$(6) \quad \bigvee_{(\forall N_i \succ N)} N_i(E) \rightarrow N(E); \quad \bigvee_{(\forall N_i \triangleright N)} N_i(E) \leftrightarrow N(E);$$

$$(7) \quad N(E) \rightarrow \bigwedge_{(\forall N_i \succ N)} N_i(E); \quad N(E) \leftrightarrow \bigwedge_{(\forall N_i \triangleright N)} N_i(E),$$

где \vee (\wedge) – логическая связка «ИЛИ» («И»); \succ (\triangleright , \triangleright , \triangleright) – знак отображения обобщения (ассоциации, типизации, агрегации) понятий; \forall – квантор всеобщности; \rightarrow – логическая связка следования.

Обратное утверждение первого правила вывода из (6) верно только для абстракции типизации-конкретизации (второе правило), так как согласно (3) абстракция обобщения-специализации описывает большее число сущностей, чем их содержится в объединении экстенционалов обобщаемых понятий.¹

В свою очередь обратное утверждение первого правила вывода из (7) верно только для абстракции агрегации-декомпозиции (второе правило), так как согласно (4) абстракция ассоциации-индивидуализации описывает меньшее число сущностей, чем их содержится в декартовом произведении экстенционалов ассоциируемых понятий.

Запросы. Для превращения информационной системы в полноценную базу знаний необходимо реализовать запросы для извлечения фактов (суждений) и вывода содержательных утверждений о моделируемой предметной области.

Факты (суждения) и утверждения представляют собой высказывания с логическими связками «И», «ИЛИ», «НЕ», в которых используются два типа предикатов:

– одноместные предикаты принадлежности сущности E понятию N вида $N(E)$;

¹ В известных реализациях баз знаний не допускается пополнение понятия-обобщения новыми сущностями кроме как через пополнение обобщаемых понятий. Это делает понятие-обобщение в некотором смысле неполноценным понятием.

– отношения вида $P[E] \circ V$, где $P[E]$ – функтор, возвращающий значение понятия-признака P сущности E , \circ – знак отношения ($=, \neq, >, \geq, <, \leq$ и т.п.), V – некоторое понятие-значение.

На естественном языке предикат принадлежности может быть выражен следующими предложениями: «Строительство есть Тип проекта», «Руководитель проекта (Главный инженер)», «Проект 12 – Учетный проект», «Проект 12, Петров Иван – это Назначение», а отношения с признаками – «Дата > 12.11.2016», «Тип проекта = Модернизация» и т.п. (рис. 8).

На рис. 16 приведена форма для поиска сущностей понятий, удовлетворяющих условиям, задаваемым с помощью шаблона поиска. Шаблон поиска состоит из списка признаков понятия и отображается в левой панели формы при установке указателя на узел поиска сущностей с пиктограммой «Ключ». После задания ограничений на значения одного или нескольких признаков информационная система выполняет поиск сущностей, удовлетворяющих заданным условиям, и выводит последние в виде дочерних узлов узла поиска.

Поиск сущностей понятия по ключу выполняется аналогично и осуществляется после ввода ключа или его части в название узла поиска. Если в ключе несколько признаков, то они разделяются знаком # (рис. 4–7). Признаки, входящие в ключ, выводятся в шаблоне поиска жирным шрифтом (рис. 16).

Для более сложного поиска необходимо использовать запросы к базе знаний, для чего предусматривается соответствующий язык запросов и поддерживающая его машина вывода на знаниях, реализующая правила вывода вида (5), (6) и (7). Однако для многих прикладных запросов оказывается достаточным использование адресной строки клиентского приложения для нахождения в понятийной структуре понятий по их именам (рис. 14), с последующим поиском требуемых сущностей в экстенционалах найденных понятий (рис. 16).¹

¹ Описание языка запросов и механизма логического вывода выходит за рамки настоящей статьи.

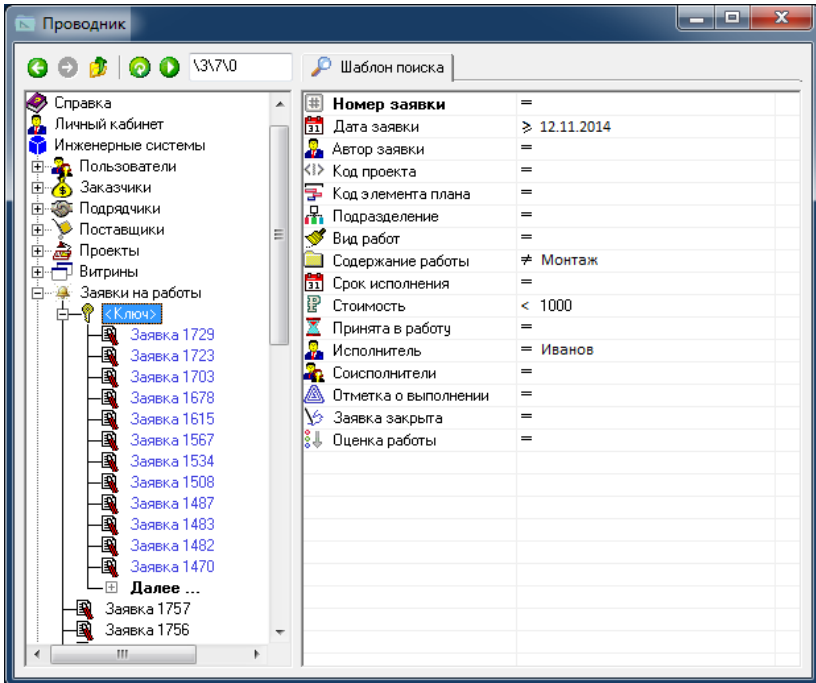


Рис. 16. Поиск в экстенционалах понятий

10. Репрезентация знаний

Помимо задач представления, извлечения и актуализации знаний имеется другая важная задача – их репрезентация. Репрезентация знаний заключается в изменении формы представления знаний и осуществляется на основе построения понятийных подмоделей с последующей их визуализацией специальными программами.

Определение 5. Подмоделью M' понятийной модели $M = (S, D)$ называется такая понятийная модель $M' = (S', D')$, для которой выполняются следующие отношения: $S' \subseteq S$, $D' \subseteq D$, где S' – некоторая подструктура (фрагмент) понятийной структуры S , D' – описание интенционалов понятий, входящих в понятийную структуру S' .

Для построения понятийной подмодели применяется следующая процедура. Вначале выделяется множество основных понятий, которые должны присутствовать в подмодели по условиям решаемой задачи. Затем итерационно строится понятийная подструктура, в которую включаются все понятия, имеющие связи с начальным, а далее – с текущим их множеством. Итерации завершаются, когда текущее множество понятий перестает пополняться. В завершении процедуры для полученной понятийной подструктуры создается описание интенционалов входящих в нее понятий.

Построение понятийных подмоделей необходимо для создания данных, которые требуются для визуализации фрагментов предметной области сторонними программами. Например, для планирования проектов часто используется понятийная модель (рис. 17), в которой присутствуют такие понятия как задача (вкладка «Задачи»), ресурс (вкладка «Ресурсы»), а также связи между задачами и назначение задачам ресурсов (вкладка «План»).

Задача	Наименование	Начало	Окончание	...
2	Блок управления шкафного исполнения или р...	13.05.2015	13.05.2015	
6	Лоток металлический штампованный по устан...	22.05.2015	22.05.2015	
3	Кабель до 35 кВ по установленным конструкц...	13.05.2015	15.05.2015	
4	Профиль перфорированный монтажный длино...	04.05.2015	04.05.2015	
8	Полка кабельная, устанавливаемая на стойка...	06.05.2015	06.05.2015	
7	Стойка сборных кабельных конструкций (без ...	11.05.2015	11.05.2015	

Рис. 17. Понятийная модель плана-графика

Для отображения такой подмодели могут использоваться формы, реализуемые соответствующими прикладными программами: диаграммы Ганта (рис. 18), ресурсные списки, графики использования ресурсов и т.п. Для этого в состав информационной системы включается модуль, выполняющий визуализацию планов-графиков на основе их понятийных моделей.

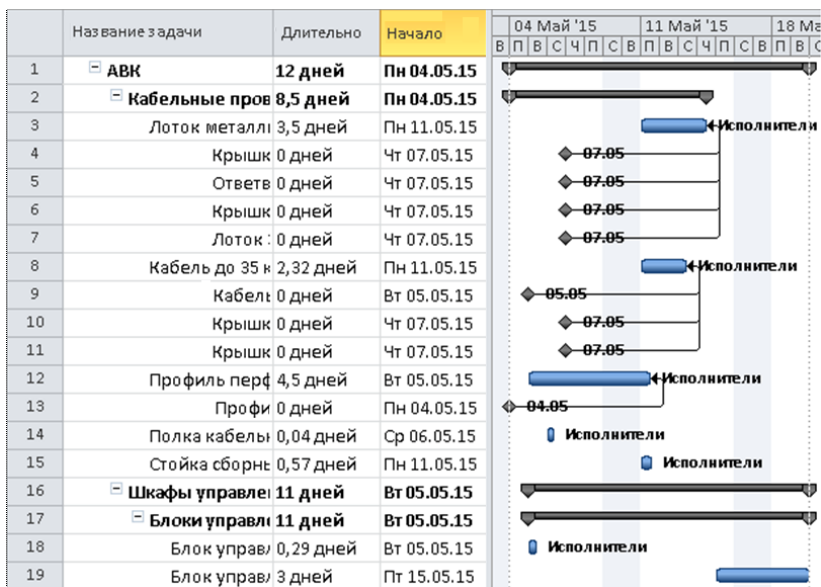


Рис. 18. Визуализация плана-графика (диаграмма Ганта)

Другим примером использования подмоделей является автоматическое создание различного рода документов (файлов) [6]. В этом случае подмодель дополняется правилами выражения (репрезентации) понятий в теле документа. Выразительные средства, используемые для такой репрезентации, будут зависеть от требуемой формы отображения (текст, графика, звук, анимация и др.).

Для отображения подмодели в текстовом виде правила репрезентации могут быть оформлены в виде шаблона документа. При создании шаблона используется специальный язык разметки, позволяющий задать формы выражения понятий в тексте. На рис. 19 приведена формальная грамматика языка текстовой репрезентации понятийных моделей.

Текст состоит из произвольных строк и операторов (правило 1). Для репрезентации понятийной модели используются операторы извлечения, вычисления, установки, выбора и итерации (правило 2).

- 1) Текст → Строка [Текст] | Оператор [Текст]
- 2) Оператор → Извлечение | Вычисление | Установка
| Выбор | Итерация
- 3) Извлечение → '[GET' Путь ['#FORMAT' Формат]]'
| ['\$' Путь ['#' Формат]]'
- 4) Вычисление → '[LET' Выражение ['#FORMAT' Формат]]'
| ['=' Выражение ['#' Формат]]'
- 5) Установка → '[SET' Путь ['#VALUE' Выражение]]'
| ['@' Путь ['#' Выражение]]'
- 6) Выбор → '[IF' Выражение '#THEN' Текст ['#ELSE' Текст]]'
| ['?' Выражение '#' Текст ['#' Текст]]'
- 7) Итерация → '[FOR' Сущность ['=' Путь] '#WHILE' Выражение
'#REPEAT' Текст]'
| ['*' Сущность ['=' Путь] '#' Выражение ['#' Текст]]'

Рис. 19. Формальная грамматика языка репрезентации

Оператор извлечения позволяет получить и вставить на место своего нахождения отформатированное значение, извлекаемое по заданному пути в понятийной модели (правило 3). Оператор вычисления используется для репрезентации в виде текста отформатированного значения некоторого вычисляемого выражения (правило 4). Синтаксис и семантика выражений – как у языков высокого уровня. В реализованной информационной системе в качестве языков для задания выражений используются интерпретируемые языки VBScript и JScript.

Оператор установки служит для изменения значений в понятийной модели и может использоваться, в том числе, для создания временных простых понятий (переменных) (правило 5). Оператор выбора необходим для реализации текстового ветвления в процессе репрезентации модели (правило 6), а оператор итерации – для репрезентации составных понятий (правило 7). Операторы могут быть вложены друг в друга, так как все части операторов представляют собой обычный текст.

На рис. 20 показан шаблон документа, а на рис. 21 – сам документ, полученный в результате репрезентации некоторой понятийной подмодели.

Поз.	Наименование и техническая характеристика	Тип, марка, обозначение документа, опросного листа
1	2	3
*prt=Specification	@prt#.Specification." & prt \$\$prt]	
*cpl=\$prt# #1@cplj #	[\$cplj.Title]? [\$cplj.Components > 0 #, в составе]	[\$cplj.Mode]
*cmp=\$cplj.Compon	@ cmpk # cplj & ".Components." & cmp [\$cmpk.Title]	[\$cmpk.Mode]
	[tbn=TableNote##=Mid("*****",1,Note("TableNote",TableNote(tbn)))] [\$TableNote(\$tbn)]	

Рис. 20. Шаблон документа

Поз.	Наименование и техническая характеристика	Тип, марка, обозначение документа, опросного листа
1	2	3
	<u>Оборудование</u>	
1	Кондиционер полупромышленный, сплит, Q=9,8 кВт, в составе:	FHQ100B / RR100BW
1.1	блок внутренний подпотолочного исполнения, однопоточный	FHQ100B
1.2	блок внешний с низкотемпературным комплектом «Иней»	RR100BW
2	Пульт управления индивидуальный, инфракрасный	ARC433A73

Рис. 21. Текстовый документ

Аналогичным образом создаются и используются подмодели других устойчивых фрагментов предметных областей и соответствующие им формы визуализации, например:

- графики и диаграммы (графическое представление данных линейными отрезками или геометрическими фигурами) [16];
- инфографика (графическое представление графов, карт, рисунков, формул и т.п.) [29];
- техническая графика (графическое представление схем, чертежей, аксонометрий) [6];

– динамические модели бизнес-процессов в различных нотациях (графическое представление процессов и их текущих состояний) [23, 32].

11. Заключение

В настоящей статье описано построение информационной системы с понятийной моделью предметной области. Модели предметной области названы понятийными, чтобы отличать их от известных концептуальных моделей. В концептуальных моделях задаются понятия (концепты) и разного рода связи (отношения) между ними, несущие часть семантической нагрузки модели. Другая часть семантики концептуальной модели содержится в дополнительных данных, доопределяющих связи между понятиями в виде логических выражений, формул, функций и т.п.

В понятийных моделях связи между понятиями сами являются понятиями, а модель строится на основе выявления и описания абстракций, послуживших образованию (определению) понятий. Отказ от описания ассоциаций в виде связей с различной семантической разметкой делает понятийную структуру предметной области представимой в виде дерева и более наглядной.

Предметная семантика полностью задается интенционалами понятий, а абстрагирование понятий, формализованное в понятийной структуре предметной области, определяет не более чем структурированность интенционалов. В этом случае не требуется задавать логические высказывания (формулы, функции), характеризующие понятия и являющиеся правилами вывода. Все, что необходимо для вывода на знаниях, содержится в понятийной структуре предметной области и интенционалах понятий.

Таким образом, коренное отличие рассмотренного подхода заключается в использовании помимо логики, еще одного семантического инварианта – правил образования и выражения

понятий.¹ Это потребовало определения ассоциаций (связей) между понятиями в виде самостоятельных понятий, а также рассмотрения обобщения как полноценного понятия, которому могут принадлежать сущности, не принадлежащие обобщаемым понятиям.

В других подходах ассоциация понятием не является, выражается связью между понятиями и определяется как единица смысла, посредством которой производится описание предметной семантики. В свою очередь обобщение не предполагает наличие собственных сущностей, которые не являются сущностями обобщаемых понятий.

Информационная система с понятийной моделью предметной области относится к классу интеллектуальных систем и предназначена для хранения и обработки знаний. Она позволяет пользователю описать предметную область: задать понятия, способы их абстрагирования, определить предметную семантику в виде интенционалов понятий, установить соответствие между предметной областью и сущностями их экстенционалов описываемых понятий, а также выполнять содержательные запросы к формируемой и постоянно изменяющейся базе знаний и репрезентацию накопленных знаний в других формах.

При таком подходе для актуализации модели при изменениях в предметной области не требуется решения задач, связанных с программированием. Более того, как актуализация модели, так и прикладные задачи предметной области решаются посредством семантически инвариантных для всех предметных областей операций создания, изменения и удаления понятий.

Другими немаловажными достоинствами информационных систем с понятийным моделированием предметной области являются:

¹ Очевидно, что процесс абстрагирования не зависит ни от какой области интерпретации, а определяется только способностями самого познающего субъекта. Следовательно, формализация способов образования и выражения понятий может рассматриваться как теория, претендующая, как и исчисление предикатов, на семантическую инвариантность во всех «мыслимых мирах».

– прозрачность – использование предельно общих и естественных методов анализа предметной области, унификация обследования предприятия перед внедрением информационной системы;

– настраиваемость – возможность учета отраслевой специфики предприятий, применимость на предприятиях любого размера и сферы деятельности, быстрота и поэтапность внедрения;

– адаптируемость – возможность формирования понятийных подмоделей для конкретных пользователей и их групп, использование единого унифицированного интерфейса пользователя, широкие возможности по настройке прав доступа к понятийной модели или ее части;

– гибкость – быстрое реагирование на изменения в предметной области, простая актуализация понятийной модели в соответствии с изменяющимися внешними условиями, легкая модифицируемость информационной системы;

– открытость – небольшое число унифицированных и устойчивых межслойных интерфейсов, способность взаимодействовать с другими информационными системами на основе семантически инвариантного языка понятийной модели;

– масштабируемость – возможность создания и использования сложных и многоаспектных понятийных моделей, расширяемость информационной системы путем увеличения числа серверов в каждом слое и динамического распределения нагрузки на серверы нижележащих слоев;

– интегрированность – легкий перенос данных от других информационных систем на основе простого и эффективного языка понятийной модели, репрезентация знаний с помощью сторонних программных средств.

Основные трудности, которые имеют место при использовании информационных систем с понятийными моделями – это необходимость освоения новой методологии и технологии моделирования предметной области и репрезентации накопленных знаний, а также отказ от устоявшихся узкоспециализированных форм пользовательского интерфейса.

В итоге, информационная система с понятийным моделированием предметной области является представителем нового поколения информационных систем в методологическом, тех-

нологическом и функциональном плане. Использование понятийной модели создает предпосылки для улучшения прозрачности бизнес-процессов предприятия, способствует оптимизации затрат и повышению инвестиционной привлекательности, уменьшает риски владения информационной системой, а именно:

- проектные риски, связанные с созданием информационной системы;
- технологические риски, связанные с потерей или искажением данных в процессе актуализации модели;
- эксплуатационные риски, связанные с поддержанием информационной системы в работоспособном состоянии и обеспечением независимости от поставщика;
- риски сопровождения, связанные с изменчивостью предметной области.

Литература

1. АБРАМОВА Н.А. *О проблеме рисков из-за человеческого фактора в экспертных методах и информационных технологиях* // Проблемы управления. – 2007. – №2. – С. 11–21.
2. БАРВАЙС ДЖ. *Теория множеств*. Справочная книга по математической логике. – Ч. 2. – М.: Наука, 1982. – 392 с.
3. ВАГИН В.Н., ГОЛОВИНА Е.Ю., ЗАГОРЯНСКАЯ А.А., ФОМИНА М.В. *Достоверный и правдоподобный вывод в интеллектуальных системах* / Под. ред. В.Н. Вагина, Д.А. Поспелова. – М.: Физматлит, 2004. – 704 с.
4. ВОЙШВИЛЛО Е.К. *Понятие как форма мышления: логико-гносеологический анализ*. – М.: Изд-во МГУ, 1989. – 239 с.
5. ВЫХОВАНЕЦ В.С. *Исчисление понятий* // Труды VII Международной конференции «Когнитивный анализ и управление развитием ситуаций» (CASC'2007). – М.: Институт проблем управления, 2007. – С. 31–35.
6. ВЫХОВАНЕЦ В.С. *Репрезентация знаний* // Материалы Международной конференции и выставки «Системы проектирования, технологической подготовки производства и управления этапами жизненного цикла промышленного продукта» (CAD/CAM/PDM'2007). – М.: Институт проблем управления, 2007. – С. 49–52.

7. ВЫХОВАНЕЦ В.С. *Прикладной понятийный анализ* // Труды VIII Международной конференции «Когнитивный анализ и управление развитием ситуаций». – М.: Институт проблем управления, 2009. – С. 62–65.
8. ВЫХОВАНЕЦ В.С. *Методы анализа крупномасштабного производства. Понятийный анализ и моделирование* // Труды III Международной конференции «Управление развитием крупномасштабных систем». – М.: Институт проблем управления, 2009. – С. 308–316.
9. ВЫХОВАНЕЦ В.С. *Мультипроблемный анализ и многоаспектное моделирование социальных систем* // Труды V Международной конференции «Управление развитием крупномасштабных систем» (MLSD'2011). – М.: Институт проблем управления, 2011. – С. 158–170.
10. ВЫХОВАНЕЦ В.С. *О понятии понятия* // Труды IX Международной конференции «Когнитивный анализ и управление развитием ситуаций» (CASC'2011). – М.: Институт проблем управления, 2011. – С. 39–42.
11. ГАСКАРОВ Д.В. *Интеллектуальные информационные системы*. – М.: Высшая школа, 2003. – 431 с.
12. ГОРСКИЙ Д.П. *Вопросы абстракции и образования понятий*. – М.: Изд-во АН СССР, 1961. – 352 с.
13. *Макетирование, проектирование и реализация диалоговых информационных систем* / Под ред. Е.И. Ломако. – М.: Финансы и статистика, 1993. – 320 с.
14. КОГАЛОВСКИЙ М.З., КАЛИНИЧЕНКО Л.А. *Концептуальное и онтологическое моделирование в информационных системах* // Программирование. – 2009. – Т. 35, №5. – С. 3–25.
15. НИКАНОРОВ С.П., НИКИТИНА Н.К., ТЕСЛИНОВ А.Г. *Введение в концептуальное проектирование АСУ: Анализ и синтез структур*. – М.: Концепт, 2007. – 236 с.
16. ПАКЛИН Н.Б., ОРЕШКОВ В.И. *Бизнес-аналитика: от данных к знаниям*. – СПб.: Питер, 2013. – 706 с.
17. РАССЕЛ С., НОРВИГ П. *Искусственный интеллект: современный подход*. – М.: Вильямс, 2006. – 1408 с.
18. СОЛСО Р. *Когнитивная психология*. – СПб.: Питер, 2002. – 592 с.
19. СТОБО ДЖ. *Язык программирования Пролог*. – М.: Радио и связь, 1993. – 368 с.
20. ФЕЙС Р. *Модальная логика*. – М.: Наука, 1974. – 516 с.

21. ЦЕЙТИН Г.С. *Программирование на ассоциативных сетях // ЭВМ в проектировании и производстве.* – Л.: Машиностроение, 1985. – Вып. 2. – С. 16–48.
22. ШЕНК Р. *Обработка концептуальной информации.* – М.: Энергия, 1980. – 360 с.
23. ЯЦУТКО А.В., ВЫХОВАНЕЦ В.С. *Динамическое управление бизнес-процессами на основе совмещенных сетей управления и данных // Инженерный журнал: Наука и инновации.* – 2013. – №2 (14). – URL: <http://engjournal.ru/catalog/mathmodel/social/530.html> (дата обращения: 09.06.2016).
24. BORGIDA A.T., CHAUDHRI V.K., GIORGINI P., et al. *Conceptual Modeling: Foundations and Applications.* – Springer-Verlag, 2009. – 527 p.
25. BRODIE M.L. *On the Development of Data Models // In: On Conceptual Modelling: Perspectives from Artificial Intelligence, Databases, and Programming Languages / Eds: M.L. Brodie, J. Mylopoulos, J.W. Schmidt.* – New York: Springer-Verlag, 1984. – P. 19–48.
26. CHEN P. *The Entity-Relationship Model – Toward a Unified View of Data // ACM Transactions on Database Systems.* – 1976. – Vol. 1, No. 1. – P. 9–36.
27. *Encyclopedia of Database Systems / Eds: L. Liu, M.T. Ozsu.* – Springer, 2009. – 3752 p.
28. GANTER B., WILLE R. *Formal Concept Analysis: Mathematical Foundations.* – Berlin: Springer, 1999. – 284 с.
29. LANKOW J., RITCHIE J., CROOKS R. *Infographics: The Power of Visual Storytelling.* – Wiley, 2012. – 264 p.
30. OLIVE A. *Conceptual Modeling of Information Systems.* – Springer, 2007. – 455 p.
31. SOWA J.F. *Knowledge Representation: Logical, Philosophical, and Computational Foundations.* – Pacific Grove: Brooks Cole Publishing, 2000. – 594 p.
32. VYKHOVANETS V., YATSUTKO A. *Dynamic business process management based on the combined control and data networks // Preprints of the 2013 IFAC Conference on Manufacturing Modelling, Management, and Control.* – Saint Petersburg: Saint Petersburg State University, 2013. – P. 672–677.

INFORMATION SYSTEM USING CONCEPTUAL DOMAIN MODEL

Valeriy Vykhovanets, Institute of Control Sciences of RAS, Moscow, Dr. Sc., Leading research associate (valery@vykhovanets.ru).

Abstract: Conceptual modeling is used for formal specification of domain knowledge and its integration into an information system. We describe an intelligent information system based on the conceptual domain modeling. The system consists of four layers: client, knowledge, logic and data. The knowledge layer is based on a conceptual model which consists of a conceptual structure and a content of concepts. A conceptual structure is specified as a set of concepts, which contains four abstractions: generalization, typification, association and aggregation. A content of each concept is specified by database tables. The significant difference between the proposed conceptual model and others is the description of the association not as a relationship but as an ordinary concept. This implies that the conceptual structure can be represented as a tree. And the information needed for knowledge-based inference is contained in the concept structure itself. We show that the semantic invariance of the conceptual interpretation improves technological and operational characteristics of the information system.

Keywords: subject area, information system, conceptual model, conceptual structure, abstraction of concepts, knowledge base, knowledge representation.

Статья представлена к публикации членом редакционной коллегии Н.И. Базенковым.

*Поступила в редакцию 05.07.2016.
Опубликована 31.03.2017.*