

# ПРОЕКТИРОВАНИЕ ОНТОЛОГИЙ НА ОСНОВЕ МОДЕЛИ ПРЕДМЕТНОЙ ОБЛАСТИ

## DESIGNING OF ONTOLOGIES BASED ON THE SUBJECT DOMAIN MODEL

*Болотова Людмила Сергеевна / Bolotova L.S.,*

*профессор кафедры «Вычислительная техника» МИРЭА(ТУ) / the professor of chair*

*"Computer facilities", MIREA(technical university)*

*lubolotova@mail.ru*

*Старых Владимир Александрович / Starykh V.A.,*

*заместитель директора ФГУ ГНИИ ИТТ «Информика» / deputy director State Institute of Information Technologies and Telecommunication*

*vstar@informika.ru*

### **Аннотация**

Предлагается подход к проектированию онтологий на основе модели предметной области, разрабатываемой методом ситуационного анализа. В его основу положены идеи ситуационного управления большими системами, концептуального анализа и проектирования систем организационного управления, теория семиотических моделей. Приводятся основные положения метода, формально-логический аппарат анализа концептуальных структур единичных решений и формирования концептуальной структуры всей предметной области, которая затем является основой для создания онтологий различных типов.

### **Annotation**

The approach to formation ontologies on the basis of model of a subject domain which designing is carried out by a method of the situational analysis is considered. Ideas of situational management are put in a basis of a method by big systems, the conceptual analysis and designing of systems of organizational management, and also the theory of semiotics models. Substantive provisions of a method, its is formal-logic device of the analysis of conceptual structures of individual decisions and formations of conceptual structure of complete model of a subject domain are resulted. Questions of automatic removal ontologies various this model are considered.

**Ключевые слова:** инженерия знаний, ситуационное управление, ситуационный анализ, модель предметной области, концептуальный анализ,

проектирование, концептуализация, формализация, решение, действие, концептуальная структура, база знаний, система управления знаниями, онтология.

**Keywords:** knowledge engineering, situational control, situational analysis, domain model, conceptual analysis, designing, conceptualization, formalization, decision, action, conceptual framework, knowledge base, knowledge control system, ontology.

## 1 Введение

Общепринятым и наиболее распространённым в настоящее время является определение онтологии, предложенное в [17], согласно которому онтология есть точная (выраженная формальными средствами) спецификация концептуализации. Под концептуализацией обычно понимается процесс перехода от представления предметной области (ПрО), выраженного на естественном языке к точной спецификации этого описания на некотором формальном языке, предполагающем компьютерную обработку. Концептуализация также трактуется как результат подобного процесса, то есть описание множества понятий (концептов) ПрО, знаний о них и связях (отношениях) между ними. С этой точки зрения онтология – это формальное представление на основе концептуализации, модели предметной области (МПрО). Сформировавшаяся в настоящее время в инженерии знаний, методология проектирования МПрО, в качестве разработки баз знаний (БЗ) для экспертных систем (ЭС), систем управления знаниями (СУЗ) и других систем, основанных на знаниях, принята и для разработки онтологий. При этом, основные проблемы инженерии знаний возникают преимущественно на первых трех этапах разработки: идентификация → концептуализация → формализация. А на этапах реализации и тестирования, соответственно, могут проявляться неполнота, некачественность описания элементов и ПрО в целом, что является причиной возврата на предыдущие этапы проектирования, и это может повторяться многократно. В этой связи проектирование МПрО до настоящего времени остаётся процессом слабоформализованным. В онтологиях проблемы извлечения знаний усугубляются повышенными требованиями к качеству семантических единиц, входящих в неё. Имеется в виду: строгость определения терминов; необходимость их принятия международным сообществом; согласованность с некоторой моделью предметной области, для которой они строились, а также – с

определённой единой моделью, принятой в данной области знаний. Существует также целый ряд требований к МПрО, определяемых спецификой онтологий. Так например, проведённое авторами статьи исследование различных глоссариев [2, 6], которые, казалось бы, очень просто и естественно можно представить в виде онтологий, показало, что определения могут оказаться противоречивыми, недостаточными или перегруженными незначущими деталями и т.п. Причина в том, что в общей проблематике создания онтологий недостаточно проработаны следующие вопросы.

- Как очертить границу предметной области для онтологии?
- Где остановить глубину описания терминов, включаемых в онтологию?
- Как построить определение термина (онтологическое соглашение по некоторому термину), отвечающее задачам онтологии – избыточное, непротиворечивое, но достаточное с точки зрения поставленных целей?
- Как определить объем и содержание контекста, необходимого для понимания выделенного термина (понятия)?
- Как наращивать онтологию при расширении предметной области без кардинальной её перестройки?
- Как научить онтологии «понимать» друг друга и т.п.?

Очевидно, что перечисленные вопросы свойственны таким онтологиям, которые по сложности выше, чем обычные таксономии.

В данной статье делается попытка ответить на поставленные вопросы с позиции теоретических основ метода ситуационного анализа и проектирования МПрО для систем поддержки принятия решений [3,4,14]. В основе предлагаемого метода, лежат идеи метода ситуационного управления большими системами [8,11,12], методологии концептуального анализа и проектирования (КАиП) систем организационного управления [9,10] и теория семиотических систем [15,16]. Рассмотрим основное содержание предлагаемого метода.

## **2 Метод ситуационного анализа и проектирования модели предметной области**

Метод ситуационного управления был разработан применительно к случаю, когда число возможных ситуаций в ПрО объекта управления существенно больше числа решений, т.е.,  $S \gg U$  [8,11,12]. Однако, как показала практика, таких систем управления сравнительно немного. Для большинства вопрос о числе возможных решений и их соотношении с числом возможных ситуаций вообще является открытым. Поэтому можно говорить о случаях, когда число возможных ситуаций либо

соизмеримо с числом решений, т.е.  $S \approx U$ , либо множество  $U$  достаточно велико и открыто, чтобы перечислить все возможные решения [3]. В качестве основного подхода к описанию внешнего мира был принят семиотический, то есть подход с позиции понятий о знаках и знаковых системах (треугольник Фреге, прямоугольник Пospelова Д.А.[11]).

В прямоугольнике Пospelова Д.А., к трём связям между знаком, концептом и денотатом (в треугольнике Фреге) добавляются ещё две, соответствующие прагматической составляющей знака:

- связь 4 – определяющая отношение между семантикой и прагматикой, то есть между смыслом ситуации и действиями над объектом, обозначаемым данным знаком; в обратном направлении - эта связь соответствует объяснению причин тех или иных событий;
- связь 5 – предполагающая переход к действиям при одном упоминании имени объекта без анализа его семантики (например, нажать на тормоз при красном сигнале светофора), а также позволяющая восстановить имя (объекта, события и т.п.) по совершенным действиям.

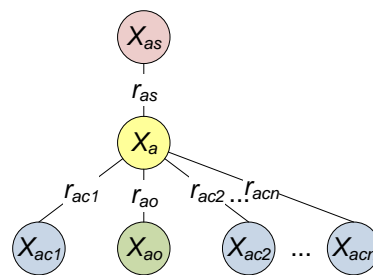
Метод ситуационного анализа рассматривает проблему построения МПрО именно для второго типа систем управления.

## ***2.1 Действия как центры кристаллизации понятий предметной области***

Человек обычно выражает свои решения с помощью предложений, в которых центральным словом, обозначающим само решение, является, чаще всего, императив в виде глагола, задаваемый в инфинитивной форме (сделать, выдать, построить, догнать, перегнать и т.п.). Все остальные элементы предложения раскрывают его смысл и отвечают на вопросы типа: кто должен выполнить действие (*субъект действия*); что (кто) является объектом этого действия (*объект действия*); как и с помощью каких средств или ресурсов это действие должно или может быть выполнено (*компоненты действия -1,2,...n*). Остальные элементы предложения выражают: либо свойства уже перечисленных, либо такие категории, как: модификаторы, квантификаторы, модальности, уточняющие области задания уже введенных элементов. Такое построение позволяет сформировать инвариантную к любой предметной области

структуру единичного решения (*decision – d*)<sup>1</sup> в виде, показанном на рис. 1а, где  $X_a$  - действие (*action – a*),  $X_{as}$  - субъект действия (*action subject – as*),  $X_{ao}$  - объект действия (*action object – ao*),  $X_{ac_1}, X_{ac_2}, \dots, X_{ac_n}$  - компоненты действия (*action components – ac*). Субъект действия, объект действия и все компоненты действия связаны с самим действием -  $X_a$  бинарными структурными отношениями  $R_{str}$  трех возможных типов:  $r_{as}(X_{as}, X_a)$  - отношение между субъектом действия и действием,  $r_{ao}(X_a, X_{ao})$  - отношение между действием и объектом действия,  $r_{ac_1}(X_a, X_{ac_1}), r_{ac_2}(X_a, X_{ac_2}), \dots, r_{ac_n}(X_a, X_{ac_n})$  - отношения между действием и компонентами действия. Эти отношения называются структурными, т.к. они связывают все части модели действия в единое целое -  $R_{str}$ .

В этой структуре четко выделяются две части: функционально - целевая (субъект – действие - объект – ФЦЧ) и обеспечивающая часть (субъект - действие - (компонента 1, компонента 2, ... , компонента n) – ОЧ)), образующие структуру единичного решения (рис. 1а). При этом ФЦЧ является целеполагающей или задачей управления  $Z$ , а ОЧ (обозначим как  $Q$ ) – перечисляет все средства, необходимые для её решения. Тогда вся структура (рис.1а) является их объединением:  $D = Z \cup Q$ , где  $Z = r_{as}(X_{as}, X_a) \cup r_{ao}(X_a, X_{ao})$ ,  $Q = r_{ac_1}(X_a, X_{ac_1}) \cup r_{ac_2}(X_a, X_{ac_2}) \cup \dots \cup r_{ac_n}(X_a, X_{ac_n})$ .



**Рис. 1а Структурная схема единичного решения**

<sup>1</sup> В статье используется англоязычная нотация, обозначающая именование индексов конструкторов, в соответствии с именованями отношений, принятая в международной практике, и более понятная пользователям современных инструментальных средств проектирования и отображения графических представлений создаваемых онтологий.

Относительно этой структуры можно сформулировать следующие утверждения:

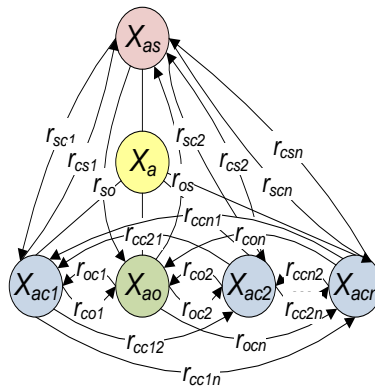
- в любом решении по управлению само решение отождествляется с действием, у которого есть субъект и объект, описывается оно как императив в виде глагола, задаваемого в инфинитивной форме (сделать, выдать, построить и т.п.);
- решение выступает в роли целеобразующей составляющей задачи по управлению и в этом смысле дерево решений может отождествляться с деревом целей или задач;
- двухуровневый граф со структурой (рис.1а) является инвариантом по отношению к произвольной ПрО.

Следуя методу КАиП, эта структура является искусственной метаструктурой, удобной для организации направленного процесса проектирования, т.е. конструктом, притом сложным, составленным из простых (терминальных) конструктов типа: субъект действия - действие (action subject - action (as-a)), действие - объект действия (action - action object (a-ao)), действие - компонента действия (action - action component (a-ac)).

Множество вершин  $d_i$  определяет объекты ПрО, необходимые для вывода данного решения, т.е., оно идентифицирует эту границу.

## **2.2 Структурные и семантические отношения**

Все вершины структуры, показанной на рис.1а, кроме вершины-действия, могут быть связаны между собой пространственными, логическими, другими типами отношений (как вертикальными, так и горизонтальными), образующими, в свою очередь, простые конструкты типа: *as-ao*, *as-ac*, *ao-ac*, *ac-ac* со своими двунаправленными бинарными семантическими отношениями -  $R_{sem}$  между входящими в них вершинами. Тогда структурная схема единичного решения превращается в представленную на рис.1.б, *семантическую структуру единичного решения* (ССЕР).

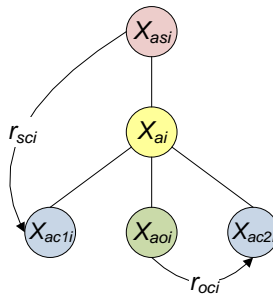


**Рис.1.6 Семантическая структура единичного решения**

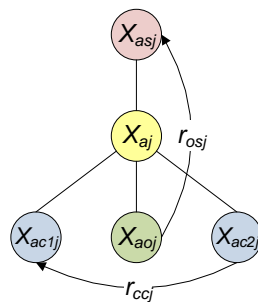
Обозначим через  $D$  - множество таких структур:

$$D = \{d_1, d_2, \dots, d_m\}, d_i = z_i \cup q_i, i = \overline{1, m},$$

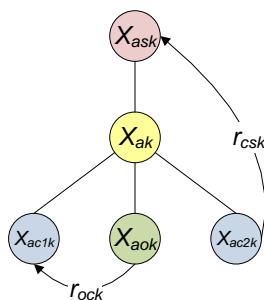
Дальнейшее изложение будем иллюстрировать простыми структурами единичных решений с двумя компонентами действия, четырьмя структурными и двумя семантическими отношениями (рис.2 а,б,в).



**Рис.2а. Структура единичного решения  $d_i$**



**Рис.2б. Структура единичного решения  $d_j$**



**Рис.2в. Структура единичного решения  $d_k$**

В итоге семантическая структура единичного решения описывается в виде:

$$d_i = \{X_{d_i}, R_{str d_i}, R_{sem d_i}\}.$$

Известно, что любое отношение для заданных объектов может рассматриваться, например, как четкая логическая переменная и принимать значения – *истина* (И), *ложь* (Л). После выявления необходимых свойств (property – p) объектов ССЕР ( $X_{d_i}$ ), ССЕР превращается в структуру, которая является уже *концептуальной структурой единичного решения* (КСЕР), так как в ней раскрывается содержание (концепт) всех входящих в неё вершин. На рис. 3а показано, что все элементы ССЕР  $d_i$  обладают своим набором свойств  $P(d_i)$ , описывающих множество её концептов –  $C(d_i)$ .

Пример условий истинности отношений между свойствами показан на рис 3.б. Например, если  $(p_{s_i}(X_{as_i}) \neq p_{c_{1i}}(X_{ac_{1i}})) = И$ , то  $\overline{РЯДОМ}(X_{as_i}, X_{ac_{1i}}) = И$ , или  $((p_{s_i}(X_{as_i}) \neq p_{c_{1i}}(X_{ac_{1i}})) = И) \rightarrow (\overline{РЯДОМ}(X_{as_i}, X_{ac_{1i}}) = И)$ . Читается это, например, так: если координаты субъекта действия  $X_{as_i}$  не равны координатам первой компоненты действия  $X_{ac_{1i}}$ , то  $X_{as_i}$  и  $X_{ac_{1i}}$  находятся *НЕ РЯДОМ*. В качестве примера на рис.3в показана ситуация:

$$((p_{s_i}(X_{as_i}) \neq p_{c_{1i}}(X_{ac_{1i}})) = И) \rightarrow (r_{sci}(X_{as_i}, X_{ac_{1i}}) = И) \text{ и}$$

$$((p_{oi}(X_{aoi}) < p_{c_{2i}}(X_{ac_{2i}})) = И) \rightarrow (r_{oci}(X_{aoi}, X_{ac_{2i}}) = И).$$



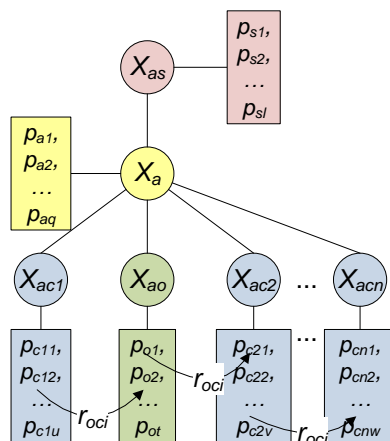


Рис. 3а. Концептуальная структура единичного решения

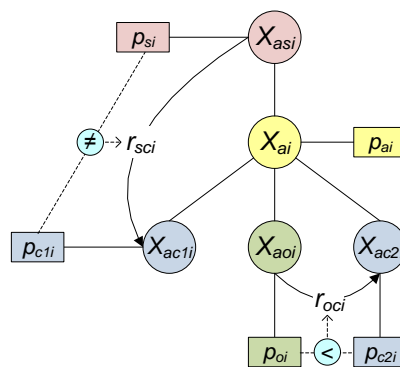


Рис. 3б. Пример условий истинности отношений в КСЕР

Соответственно, модель фрагмента ПрО единичного решения расширяется за счет добавления множества свойств объектов и отношений между ними, т.е.:

$$C(d_i) = \{X_{d_i}, R_{str d_i}, R_{sem d_i}, P_{d_i}, R_{p d_i}\}, \quad i = \overline{1, m},$$

где:  $P_{d_i}$  – свойства всех объектов названных в КСЕР  $d_i$ ,  $R_{p d_i}$  – отношения между свойствами объектов  $d_i$ .

Общепринятым является такой факт, что составными компонентами интеллектуальных систем являются модели предметной и проблемной областей, связанные с интенциональным и экстенциональным представлениями. В экстенциональную часть входят конкретные факты, касающиеся предметной ПрО, а в интенциональную – схемы связей между данными, используемых для описания факта связей между данными. Экстенциональные представления описывают конкретные объекты из ПрО, конкретные события, происходящие в ней, или конкретные явления и

процессы, а интенциональные – фиксируют те закономерности и связи, которым описываемые компоненты ПрО должны удовлетворять в рамках данной ПрО. В нашем случае,  $C_{d_i}(X_{d_i})$  представляет интенциональную составляющую КСЕР  $d_i$  -  $Int_{d_i}$ , или схему, используемую для описания факта связей между данными в КСЕР  $d_i$ , и фиксирует те закономерности и связи, которым описываемые компоненты должны удовлетворять в рамках данной предметной области. Соответственно, в экстенциональную составляющую КСЕР ( $Ext_{d_i}$ ) войдут конкретные факты, касающиеся предметной области или её части.

Можно сказать, что понятия классов, подклассов, экземпляров классов, где понятие *класса* ассоциируется с совместно используемыми свойствами, представляют собой способы организации фактов.

Пусть  $X_{d_i} = (X_1, X_2, \dots, X_v)$  – множество классов объектов действия  $d_i$ . Любой элемент  $X_i$  из этого множества обладает своим концептом представимым в виде реляционного отношения:

$C(X_i) = (p_{i1}(DOM(p_{i1})), p_{i2}(DOM(p_{i2})), \dots, p_{iw}(DOM(p_{iw}))), i = \overline{1, n}$ , где  $X_i$  – имя  $i$ -го класса объектов;  $p_{i1}, p_{i2}, \dots, p_{iw}$  – имена свойств  $i$ -го класса объектов (атрибутов),  $DOM(p_{ij})$  – домен, т.е. область значений  $j$ -го свойства  $i$ -го класса объектов ( $p_{ij}$ ),  $j = \overline{1, w}$ .

Такое описание задает статическую, или *интенциональную (внутреннюю)* составляющую концепта  $X_i$ . Таким образом, понятия *интенционала* (INT – INTensional) и *концепта* являются синонимами и определяются как:

$Int(X_i) = C(X_i) = \bigcup_{j=1}^m p_{ij}(DOM(p_{ij})), i = \overline{1, n}, j = \overline{1, m}$ . Соответственно, концептуальная

модель множества классов объектов КСЕР  $d_i$  ( $X_{d_i} = (X_1, X_2, \dots, X_v)$ ) – является

объединением их интенционалов, т.е.:  $C(X_{d_i}) = \langle \bigcup_{i=1}^w Int(X_{d_i}) \rangle$  Поскольку  $X_i$  является

классом объектов, т.е.,  $X_i = (X_{i1}, X_{i2}, \dots, X_{iv})$ , то все его экземпляры  $i = \overline{1, v}$  наследуют свойства своих классов (с возможными исключениями, специально указываемыми, со значениями, входящими в определенные выше области значений (домены)). Понятие концепта КСЕР  $d_i$  уточняется тогда, как:

$$C_{d_i}(X_{d_i}) = \left\{ \bigcup_{i=1}^n C(X_{d_i}), R_{str_{d_i}}, R_{sem_{d_i}}, R_{p(d_i)} \right\}$$

В общем случае значения свойств экземпляра класса  $X_i$  задаются фактами  $F_1, F_2, \dots, F_k$ , со структурой: *имя экземпляра класса - имя свойства – значение свойства* и задают экстенциональную (внешнюю) составляющую  $d_i$  или МПрО в целом.

Соответственно, экстенсионалом (*EXT – EXTensional*) класса объектов  $X_i$  называется выражение:

$$EXT(X_i) = \{F_1, F_2, \dots, F_k, \dots, F_h\},$$

где  $F_k = (X_i \langle p_{ij}, a_{ijk} \rangle)$ , т.е.  $F_k$  – есть факт, описывающий  $j$ -й атрибут ( $p_{ij}$ ) класса объектов  $X_i$  парой из  $p_{ij}$  (имени свойства-атрибута) и  $a_{ijk}$  – значения  $j$ -го атрибута класса  $X_i$ . Объединение интенсиналов множества классов объектов  $X$  и их экстенсиналов, получается полное описание множества классов объектов КСЕР  $d_i$ :

$$X_{d_i} = \langle \bigcup_{i=1}^v (INT(X_{d_i}), EXT(X_{d_i})) \rangle.$$

### 2.3 Концептуальная модель предметной области

Для проектирования полной КМПрО, разрабатываются КСЕРы для всех элементов множества  $D = \{d_1, d_2, \dots, d_m\}$ . Итоговая концептуальная модель предметной области (КМПрО), соответственно, является объединением концептов (интенсиналов)

множества всех КСЕР заданной ПрО:  $C_{МПРО}(X) = \left\{ \bigcup_{i=1}^n C(X_{d_i}), R_{str}, R_{sem d_i}, R_{p(d_i)} \right\}$ . Однако, это

объединение не механическое, т.к. над  $d_i$  должен быть выполнен ряд операций аналогичных таким теоретико-множественным, как пересечение, дополнение, разность, объединение и др. Кроме того, между КСЕРами может существовать отношение части и целого в случаях, когда единичное решение может быть развернуто в дерево решений, аналогичное дереву разбиения задачи на подзадачи. Графически такое решение описывается в виде графа редукции (И\ИЛИ-графа), вершины которого отождествляются с именами решений - действий –  $X_{a_i}$  ( $i = \overline{1, n}$ ), а дуги – с отношением *часть – целое* (*part of while – r<sub>pow</sub>*) или связи типа *действие – поддействие*.

Определим основные операции над КСЕРами  $d_i, d_j$ .

**Разность**  $d_i, d_j$  определяется как  $d_{ij} = d_j \setminus d_i$ ,  $i, j = \overline{1, m}$ , т.е.  $d_{ij}$  - та часть  $d_j$ , которая отсутствует в  $d_i$ , и, соответственно, концепт  $d_j$  - это разность их концептов :

$$C_{d_{ij}}(X_{d_{ij}}) = C_{d_i}(X_{d_i}) / C_{d_j}(X_{d_j})$$

**Объединение**  $d_i, d_j$  – это

$d_{ij} = d_i \cup (d_j / d_i), \quad i, j = \overline{1, m}$  с концептом:  $C_{d_{ij}}(X_{d_{ij}}) = C_{d_i}(X_{d_i}) \cup (C_{d_i}(X_{d_i}) / C_{d_j}(X_{d_j}))$ ,  
 $i, j = \overline{1, m}$ .

**Пересечение**  $d_i, d_j$  определяется как  $d_{ij} = d_i \cap d_j, \quad i, j = \overline{1, m}$ , т.е.  $d_{ij}$  – это часть  $d_i, d_j$ , являющаяся общей для них. Соответственно, концепт  $d_{ij}$  определяется как:  $C_{d_{ij}}(X_{d_{ij}}) = C_{d_i}(X_{d_i}) \cap C_{d_j}(X_{d_j})$ .

Возможны пересечения следующих типов:

1.  $d_i \cap d_j = \emptyset$ .

2. При  $d_i \cap d_j \neq \emptyset$  возможны следующие варианты пересечений.

2.1. Пересечение по одному типу вершин.

2.1.1. Пересечение по субъекту действия:  $d_i \cap d_j = X_{as_{ij}}, \quad i, j = \overline{1, m}$ .

2.1.2. Пересечение по самому действию:  $d_i \cap d_j = X_{a_{ij}}, \quad i, j = \overline{1, m}$ .

2.1.3. Пересечение по объекту действия:  $d_i \cap d_j = X_{ao_{ij}}, \quad i, j = \overline{1, m}$ .

2.1.4. Пересечение по одной или нескольким компонентам действия:

$d_i \cap d_j = X_{c_{1ij}} \wedge X_{c_{2ij}}, \quad i, j = \overline{1, m}$ .

2.2. Аналогично, могут быть рассмотрены, важные для организации логического вывода решений пересечения по двум типам вершин (по субъекту действия и самому действию:  $d_i \cap d_j = X_{as_{ij}} \wedge X_{a_{ij}}, \quad i, j = \overline{1, m}$ ; по субъекту действия и объекту действия:

$d_i \cap d_j = X_{as_{ij}} \wedge X_{ao_{ij}}$ ; по субъекту действия и одной или нескольким компонентам

действия:  $d_i \cap d_j = X_{as_{ij}} \wedge X_{c_{1ij}} \wedge X_{c_{2ij}}$ ; по действию и объекту действия:  $d_i \cap d_j = X_{a_{ij}} \wedge X_{ao_{ij}}$ ;

по действию и одной или нескольким компонентам действия:  $d_i \cap d_j = X_{a_{ij}} \wedge X_{ac_{1ij}} \wedge X_{ac_{2ij}}$ ;

по объекту действия и одной или нескольким компонентам действия:

$d_i \cap d_j = X_{ao_{ij}} \wedge X_{c_{1ij}} \wedge X_{c_{2ij}}, \quad i, j = \overline{1, m}$ .

2.3. Таким же образом могут быть определены пересечения по трем типам вершин:

по субъекту действия, самому действию и объекту действия:  $d_i \cap d_j = X_{as_{ij}} \wedge X_{a_{ij}} \wedge X_{ao_{ij}}, \quad i, j = \overline{1, m}$ ;

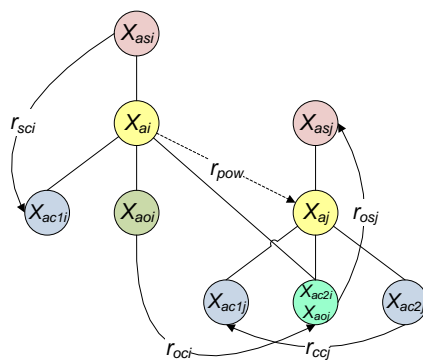
по субъекту действия, самому действию и одной или нескольким компонентам действия:  $d_i \cap d_j = X_{as_{ij}} \wedge X_{a_{ij}} \wedge X_{c_{1ij}} \wedge X_{c_{2ij}}$ ;

по субъекту действия, объекту действия и одной или нескольким компонентам действия:

$d_i \cap d_j = X_{asi} \wedge X_{aoi} \wedge X_{c1ij} \wedge X_{c2ij}$ ; по действию, объекту действия и одной или нескольким компонентам действия:  $d_i \cap d_j = X_{aj} \wedge X_{aoj} \wedge X_{ac1j} \wedge X_{ac2j}$ .

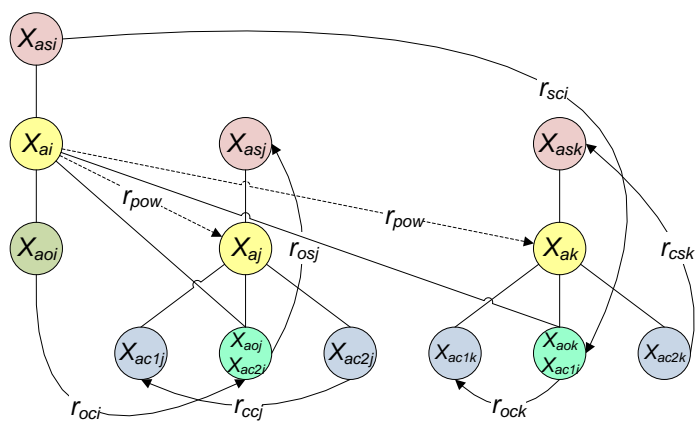
После выполнения этих операций над КСЕРами формируются полные описания всех элементов структур действий одного уровня в виде концептов субъектов, объектов, компонент действий, самих структур действий.

Одна или несколько компонент действия  $d_i$  могут сами являться объектами одного или нескольких действий в других КСЕРах, т.е. объекты - компоненты могут играть одновременно в данной ПрО несколько ролей и для каждой из них важны разные свойства и отношения, т.е. у них могут быть разные концепты. При этом, действие  $X_{aj}$ , очевидно, не может быть выполнено, пока не будет выполнено требуемое действие над компонентой  $X_{ac2i}$  действия  $X_{ai}$ . Это означает, что между действиями структур  $d_i, d_j$  автоматически проявляется неявное отношение *действие – поддействие* (рис.4). Если таких компонент несколько, то вершины-действия структур  $d_j, d_k$  ( $j, k = \overline{1, m}$ ), связаны отношением редукции *ИЛИЛИ*. Для случая *ИЛИ* необходимо устанавливать приоритеты выполнения альтернативных действий. Для случая *И* вопросы пересечения структур  $d_j, d_k$ , рассматриваются аналогичным 2.1.1-2.1.3 образом.



**Рис. 4. Пример проявления неявного отношения иерархии между КСЕР  $d_j, d_k$**

Объектами других действий могут быть одна или несколько компонент действия. Рассмотрим их взаимодействие, предположив, что две компоненты действия  $d_i$  являются, в свою очередь объектами действий  $d_j, d_k$  (рис.5).



**Рис. 5. Пример КСЕР, когда компоненты одного действия являются объектами других действий**

Здесь тоже возможны варианты взаимодействия разноуровневых структур.

1. Субъекты действия в структурах  $d_i$ ,  $d_j$  не совпадают, т.е.,  $X_{as_i} \neq X_{as_j}$ . Это означает, что эти действия выполняются разными субъектами, между которыми должно существовать отношение организационной иерархии (подчинения - рис.6,7). Следовательно, требование наличия организационного подчинения проявляется автоматически. Это отношение естественно интерпретировать как отношение «часть-целое» ( $r_{pow}$ ).

Допустим также, что для  $X_{as_i}$  и  $X_{as_j}$  построены соответствующие КСЕРы -  $d_i$ ,  $d_j$  и их концепты:  $C_{d_i}(X_{d_i}), C_{d_j}(X_{d_j})$ .

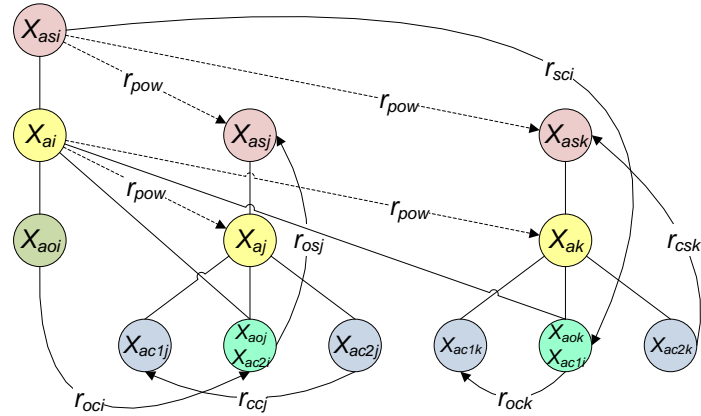
Соответственно, концепт  $d_{ij}$ :

$$C_{d_{ij}}(X_{d_{ij}}) = C_{d_i}(X_{d_i}) \cup C_{d_j}(X_{d_j}) \cup \wedge r_{pow}(X_{as_i}, X_{as_j}) \cup r_{pow}(X_{ao_i}, X_{ao_j}).$$

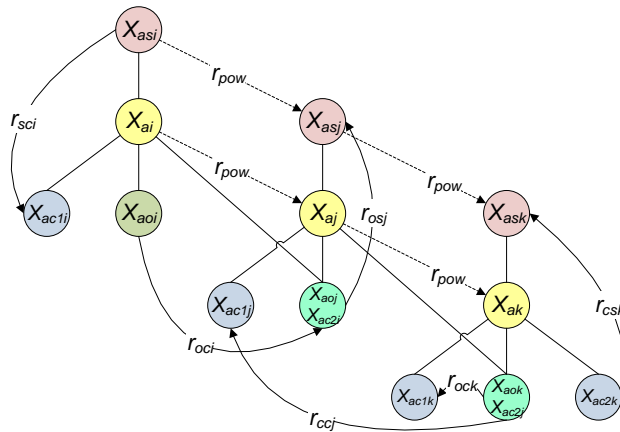
Цепочка вложенных КСЕР, образуемая по отношениям *действие - поддействие*, *субъект - надсубъект*, *объект - подобъект* может продолжаться до тех пор, пока не будет достигнут уровень терминальных действий-решений, т.е. элементарных действий, выполняемых за один шаг. Уровни терминальности действий в разных системах управления могут различаться в зависимости от уровня автоматизации в них процессов управления. В результате получается концепт (концептуальная структура) решений всей ПрО (КСРПрО) вида:

$$D = \bigcup_{i=1}^m d_i \cup G(X_{as}, R_{pow}) \cup G(X_a, R_{pow}) \cup G(X_{ao}, R_{pow}),$$

где  $G(X_{as}, R_{pow})$ ,  $G(X_a, R_{pow})$ ,  $G(X_{ao}, R_{pow})$  – графы-деревья иерархии: субъектов действий (субъектный уровень описания); самих действий (операциональный уровень описания); объектов (объектный уровень описания), соответственно.





**Рис. 6. Пример производной структуры, когда субъекты действий  $d_i, d_j, d_k$  не совпадают**



**Рис. 7. Пример производной структуры, когда субъекты действий  $d_i, d_j, d_k$  и сами действия образуют иерархии**

Концептуальная структура решения (КСР), являющегося корнем дерева решения ПрО, таким образом, представляется в виде трехслойного графа (рис. 8): первый слой – иерархия субъектов действий; второй слой – иерархия действий, третий слой описывает уровень взаимодействия объектов, компонентов действий и отношений между ними.

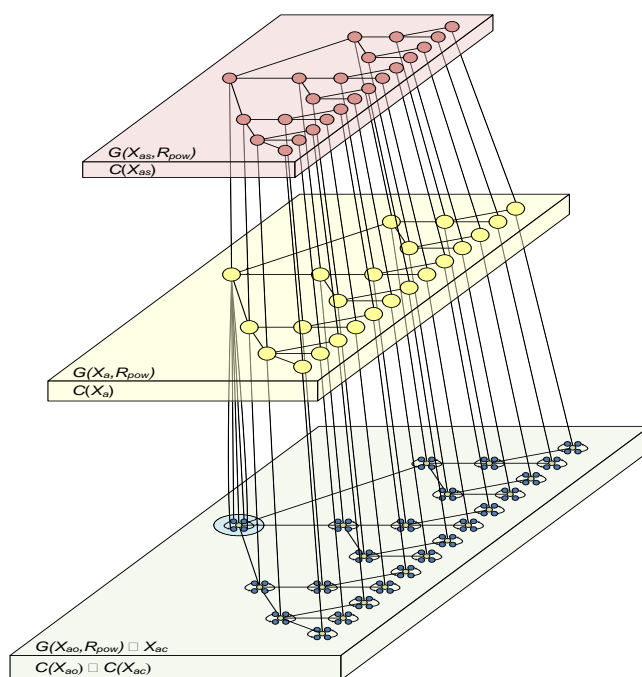
Указанный граф определяет концептуальную МПрО в целом (КМПрО). На этом рисунке: графический символ вида  изображает множество компонент, обеспечивающих семантические отношения между объектами и компонентами

действий. Пунктирными вертикальными линиями показаны структурные отношения между парами:  $(X_{as} - X_a), (X_a - X_{ao}), (X_a - X_{ac})$ ; пунктирные круги вокруг графических символов вида  – группы семантических отношений в рамках каждой  $d_i$  (для упрощения рисунка показан только один такой графический символ, а они должны быть вокруг всех вершин типа  $X_{oc}$ ); сплошными линиями показаны отношения иерархии на уровне субъектов действия, действий и компонент действий.

На рис 8 не показан ещё один уровень, соответствующий уровню свойств всех элементов МПрО и отношений между ними.

В целом, КМПрО описывается выражением:

$$G(D) = \bigcup_{i=1}^m C(d_i) \cup G(X_{as}, R_{pow}) \cup G(X_a, R_{pow}) \cup G(X_{ao}, R_{pow}),$$



**Рис. 8. Концептуальная модель предметной области**

Граф  $(G(D))$ , образующийся в результате сопоставления действиям соответствующих КСЕР, по существу, задает все множество путей (или сценариев) многошагового вывода решений. Вершины третьего типа на своем уровне связываются как отношениями иерархии, так и семантическими отношениями типа: место, средство, способ и т.п. Назовем первый уровень графа – субъектным, второй – операциональным, а третий – объектным. Каждый из этих слоев описывается своим графом: первый и



второй уровни имеют один и тот же вид – вид дерева и различаются лишь интерпретацией вершин; третий является семантической сетью, в которой каждая дуга имеет своё имя. Операциональный граф даёт полное представление об иерархии действий, их перечне на каждом уровне принятия решений и соответствует классификатору действий, описываемому в виде:  $D = (X_a, r_{pow})$ , где  $X_a$  – множество действий;  $r_{pow}$  – отношение «часть-целое», интерпретируемое здесь как отношение действие-поддействие. Число уровней классификатора действий зависит от глубины конкретизации, необходимой для достижения цели, определяемой корнем дерева, и достигнутой в исследуемой ПрО степени автоматизации процессов управления. Очевидно, что чем он выше, тем этих уровней меньше.

Объектный граф представляет собой семантическую сеть со структурными и семантическими отношениями. По отношениям между вершинами сети всегда можно установить все роли, в которых может выступать каждый объект. Очевидно, что каждой вершине классификатора действий ( $X_{a_k}$ ) соответствует свой подграф вывода –  $G_k(d_k)$ , ( $k = \overline{1, m}$ ), который, по существу, образует сценарий вывода решения, содержащий КСЕРы всех действий, лежащих на пути к целевой вершине. Таким образом, граф  $G(D)$  можно рассматривать как множество возможных сценариев вывода алгоритмов решений  $G_k(d_k)$ , – соответствующих  $d_k$ -ой вершине КСР  $G(D)$ . Порядок вывода может быть как сверху-вниз (обратная стратегия вывода «от цели - к ситуации»), так и снизу-вверх (прямая стратегия вывода «от ситуации - к цели») и регламентируется классификатором действий.

Построением графа  $G(D)$  завершается процесс формирования КМПрО и структуры соответствующего ситуационного пространства ПрО, в котором есть все необходимые знания для принятия решений на любом уровне конкретизации.

Метод ситуационного анализа реализован в виде технологии извлечения знаний, поддерживается программно и гарантирует получение качественного результата на выходе в виде БЗ ПрО, а также концептуальной модели самой предметной области. В эту технологию естественным образом вплетаются знания экспертов или их групп, регламенты субъектов управления, знания из текстов, системы классификации и т.п. [4,5,15-18]. Предложенный метод прошел многолетнюю апробацию при проектировании реальных экспертных систем поддержки принятия решений (ЭСППР),

внедрен в учебный процесс в ряде вузов для выполнения лабораторных, курсовых и дипломных работ. Накоплен огромный банк учебных и прикладных ЭСППР [5].

### **3. Создание онтологий на основе концептуальной модели предметной области**

С точки зрения требований концептуализации, определены следующие типы онтологий[1,7,18,19]:

- *Онтологии верхнего уровня*, описывающие общие понятия такие, как пространство, время, материя, объект, событие, действие и т.п., которые не зависят от конкретной задачи в ПрО.

- *Онтологии представления*, определяющие концептуализацию, которая лежит в основе формализма представления знаний. Понятия и отношения, определенные в других видах онтологий, считаются конкретизацией понятий онтологии данного вида. Предполагается, что они не зависят от конкретной ПрО.

- *Промежуточные онтологии*, содержащие общие понятия и отношения, характерные для конкретной предметной области. В идеальном случае, они используются в качестве интерфейса между онтологиями предметных областей и общими онтологиями, но могут выступать как онтологии верхнего уровня для описания знаний конкретной ПрО.

- *Онтологии предметных областей*, содержащие понятия определенной области знаний или входящих в нее областей; состоят из объектов и связей между ними, описанных в терминологии конкретной предметной области.

- *Онтологии задач*, описывающие определенные задачи или деятельность, используя словарь, введенный в общих, промежуточных онтологиях и онтологиях предметных областей; ориентированы на решение конкретных проблем и включают все понятия, необходимые для описания процесса логического вывода, от самых абстрактных понятий, относящихся к схеме вывода, до более специальных, характерных для отдельных методов.

- *Онтологии-приложения*, являющиеся специализацией онтологий предметных областей и онтологий задач и опирающиеся на определения, характерные для конкретного приложения; описывают понятия, зависящие как от конкретной проблемной области, так и от задачи; эти понятия часто характерны для обеих соединяемых онтологий и соответствуют ролям, играемым объектами предметной области при выполнении определенной деятельности.

Онтология предметной области (или ее часть) может образовывать концептуальные связи с несколькими ПрО. Одно и то же множество объектов с одним и тем же набором свойств может использоваться в различных ПрО, для каждой ПрО характерна определенная модель поведения объектов предметной области. Знания предметной области вместе с моделью, описывающей их поведение в рамках определенных в онтологии задач функций, образуют частную модель ПрО. Программная реализация данной модели является онтологией-приложением. Онтологию-приложение можно рассматривать как усеченную онтологию ПрО применительно к включенным в онтологию-приложение задачам.

Таким образом, все перечисленные способы организации онтологий предметной области ориентируются на выполнение негласно существующего соглашения в области искусственного интеллекта: в реальном мире есть *объекты*; у них есть *свойства*, которые имеют *значения*; объекты могут быть в различных отношениях друг с другом; свойства и отношения изменяются во *времени*; в различные *моменты времени* возникают *события*. Кроме этого - существуют *процессы*, в которых участвуют объекты и которые возникают во времени; мир и его объекты могут находиться в различных *состояниях*; события могут вызывать другие события, т.е. могут давать *эффект*; объекты могут состоять из *частей*. Этим понятиям соответствуют способы организации фактов мира в *классы*, *экземпляры классов* и *подклассов* и т.п., которые совместно с отношениями типа *часть - целое* (*is-a* и др.), являются фундаментальными для представления онтологий.

Анализируя КМПрО, построенную в соответствии с технологией метода ситуационного анализа (МСА) видно, что эта модель содержит всё, что должна содержать онтология уровня выше, чем таксономия. В данном случае справедливы следующие утверждения.

- Концептуальная модель ПрО (КМПрО), сформированная МСА (рис.8), содержит всё множество классов (их имен и соответствующих им концептов) онтологий предметных областей, ориентированных на решение задач управления и поддержки принятия решений (ППР). Они являются вершинами иерархий (субъектов, действий (процессов) и/или объектами соответствующих КСЕР).

- Все концепты классов объектов КМПрО содержат необходимый и достаточный набор свойств (атрибутов) вместе с необходимыми ограничениями на область их задания для решения множества задач заданной предметной области или части задач, ориентированных на реализацию отдельных функций (подфункций)

управления и ППР.

- КМПрО содержит все множество отношений между объектами ПрО и условий (аксиом) для определения значений их истинности, необходимых и достаточных для логического вывода решений.

- КМПрО содержит ответы на типы запросов, характерных для онтологий. Это запросы типов: «Кто?», «Что?», «Когда?», «Почему?», «Зачем?», «Каким способом?», «С помощью чего?», «Как?», «Какого рода (вида, подвида)?», «Что нужно сделать для того, чтобы?», «Какими свойствами обладает?», «Какими отношениями связаны два или более объектов?» и т.п.

- КМПрО в потенциале содержит множество всех онтологий задач, онтологий-приложений для заданной предметной области.

Отметим следующие свойства КМПрО, важные, на наш взгляд, для получения по ней онтологии:

- КМПрО проста для понимания и для выявления нужных понятий.
- КМПрО, как онтология, хорошо структурирована.
- КМПрО, как онтология, содержит концептуальные, а не эпизодические знания.

- На основе КМПрО естественным образом могут быть сформулированы точные аксиомы вывода, согласованные со структурой, именами и содержимым для всех определенных в ней понятий.

- Над элементами КМПрО по аналогии с онтологиями, введена алгебра (пересечение, объединение (соединение), разность. Этот список может быть расширен до необходимого для работы с онтологиями.

#### **4. Вопросы проектирования онтологий**

Рассмотрим, как КМПрО отвечает на перечисленные во введении вопросы.

##### *1. Определение границ предметной области для онтологии.*

Граница ПрО фиксируется иерархией действий – поддействий (задач – подзадач, целей - подцелей), определяемых на этапе идентификации онтологии. Сами формулировки, например, целей могут носить сколь угодно как общий, так и конкретный характер. Например, построить онтологию проблемы управления инцидентами инфраструктурой корпоративной информационной системы. В данном

случае действие *управлять* может рассматриваться как совокупность поддействий (подзадач) с соответствующими КСЕРами: *Идентифицировать инцидент (событие); Определить причину инцидента; Обработать инцидент (разрешить)*. Поддействие *Обработать*, в свою очередь, распадается на ряд своих поддействий, в зависимости от сложности инцидента и т.д.

При этом, если нужна только онтология предметной области без функций управления, в итоговой модели может исключаться второй (операциональный) уровень описания (рис.8).

## 2. *Выбор глубины описания терминов, включаемых в онтологию.*

Ответ на этот вопрос очевиден. Глубина (детальность) описания ПрО определяется терминальным уровнем действий, т.к. предполагается, что они могут быть реализованы за один шаг, например, нажатием на кнопку управления и т.п.

## 3. *Проектирование определений терминов.*

Этот вопрос заслуживает особого внимания и является предметом отдельного исследования. Отметим только следующее. В итоговой КМПрО формируются концепты всех участвующих в ней объектов (их имен как знаков). Концепт – это множество существенных свойств (признаков, атрибутов) объекта (знака), на основе которых он однозначно может быть идентифицирован во внешнем мире. В данном случае, речь идёт о множестве свойств, необходимых для решения всех подзадач данной предметной области. Кроме того, известны все отношения выделенного объекта с другими из данной ПрО. Таким образом, в определении объекта, как термина, должны присутствовать все необходимые с точки зрения онтологии свойства и ссылки на другие объекты через соответствующие отношения с ними. Это определение будет иметь всё необходимое и достаточное для достижения поставленных перед онтологией целей.

## 4. *Определение объема и содержания контекста для понимания выделенного термина.*

Контекст понятия может рассматриваться, исходя из положения соответствующего объекта в объектной иерархии (объектный уровень КМПрО на рис.8) и его множества компонент, обеспечивающих семантические отношения между объектами и компонентами действий. Это множество, рассматриваемое как ближайшее окружение объекта, соответственно, может иметь радиус достижимости равный: 1 (соответствует первому, ближайшему поддействию); равный 2 (второе по достижимости, поддействие) и т.п.

## 5. *Расширение онтологии.*

Расширение онтологии связано с расширением задач (целей) её использования. Можно предположить, что это может привести к расширению операционного графа в ширину (И\ИЛИ), в глубину и появлению новых вершин, свойств и отношений на соответствующих уровнях. Сложность такой корректировки, естественно, зависит от того, на каких уровнях будут изменения – уровни класса, подкласса, экземпляра класса, простого расширения области значений свойств и отношений. Все проблемы по уровню сложности могут быть соотнесены с проблемами модификации фреймовых структур. Но проблемы эти – чисто технического плана.

## 6. *Как научить онтологии «понимать» друг друга*

Это, пожалуй, самый сложный вопрос, поскольку связан с проблемой стандартизации и унификации определений терминов, а ещё проще, - с проблемой построения единого понятийного пространства в рамках международного сообщества. Решение этой проблемы развивается по двум направлениям – технологическому и методологическому. Технологическое направление развивается наиболее активно в части создания инструментальных средств и специализированных языков онтологий. Инструментальные средства сейчас представлены достаточно широким набором редакторов онтологий Ontolingua, Protege, OntoEdit, OilEd, WebOnto, ODE, KADS22. [7], с типовыми функциями проектирования и сопровождения онтологий, служащих для редактирования, просмотра, документирования онтологий, импорта и экспорта онтологий между системами. В ряде систем (Chimaera, OBSERVER, OntoMerge, PROMPT), кроме функции редактирования онтологий, предусмотрено выполнение таких операций над онтологиями, как *выравнивание* — установка различного вида соответствий между двумя онтологиями для того, чтобы они могли использовать информацию друг друга; *отображение* — нахождение семантических связей между подобными элементами разных онтологий; *объединение* — формирование онтологии, объединяющей информацию двух или более заданных онтологий. Для описания онтологий применяются различные языки онтологий [20], к которым относятся как традиционные языки спецификации онтологий (например, Ontolingua, CycL); языки, основанные на дескриптивных логиках (LOOM); языки, основанные на фреймах (OKBC, OCML, Flogic) и языки, основанные на Web-стандартах (XOL, SHOE, UPML). Специально для обмена онтологиями через Web разработаны и используются языки RDF, DAML, OIL, OWL.

Методологическое направление решения этой проблемы, в первую очередь связано с необходимостью унификации таксономий. Так например, многие научные термины, составляющие таксономию имеют синонимы, число которых у некоторых таксонов достигает десятков. Кроме того, для каждой предметной группы таксонов существуют, как правило, параллельно несколько альтернативных таксономических систем, определяющих различные представления о числе этих таксонов и о структуре их иерархических связей. Если рассматривать решение данной проблемы с точки зрения международной связности и требований открытости, то ситуация ещё более усугубляется необходимостью понятийных согласований терминов и определений на уровне многоязычного представления. Видимо, здесь будет очень важна инициатива международных групп разработчиков онтологий в части формализации обобщённых представлений, с последующим обсуждением и принятием онтологических стандартов.

### *Литература*

1. Башмаков А.И., Башмаков И.А. Интеллектуальные информационные технологии. М. МГТУ им. Баумана, 2005, с.- 302.
2. Башмаков А.И., Старых В.А. Систематизация информационных ресурсов для сферы образования. - М.Фонд «Европейский центр по качеству», М. 2003, 383 с.
3. Болотова Л.С., Иноземцев С.П., Яценко Н.Ю. Модели и методы представления знаний в системах искусственного интеллекта. - СПб.: ВИКИ им. А.Ф. Можайского, 1991. - 116 с.
4. Болотова Л.С., Смольянинова В.А., Смирнов С.С. Теоретические основы концептуального проектирования модели предметной области в корпоративных информационных системах // Теоретические вопросы вычислительной техники и программного обеспечения: Межвуз. сб. научн. тр. - М.: МИРЭА, 2006. - С. 125-131.
5. Болотова Л.С., Смольянинова В.А., Смирнов С.С. Понятийно-объектная модель как формализованное представление концептуального пространства предметной области // 51-я научно-техническая конференция: Сб. тр. - М.: МИРЭА, 2002. - С. 86-90.
6. Глоссарий терминов и определений (Glossary Terms and Definitions. ITIL V3 Glossary, Russian Translation, 30 April, 2009 V0. - перевод Жилинского А., Сарычева А.
7. Иванников А.Д., Кулагин В.П., Мордвинов В.А. и др. Получение знаний для формирования информационных образовательных ресурсов. М.: ФГУ ГНИИ ИТТ «Информика, 2008.-440 с.
8. Клыков Ю.И. Ситуационное управление большими системами. - М.: Энергия,

1974. - 135 с.

9. Кучкаров З.А., Никаноров С.П., Солнцев Г.В., Шабаров В.Н. Концептуальный анализ и проектирование. Исследование социально-экономических систем. Методология. Теория. Следствия. - М.: Концепт, 2007 - 844 с.
10. Никаноров С.П., Никитина Н.К., Теслинов А.Г. Введение в концептуальное проектирование АСУ: Анализ и синтез структур. - М.: РВСН, 1995. - 234 с.
11. Поспелов Д.А. Ситуационное управление: Теория и практика. - М.: Наука, 1986. - 284 с.
12. Поспелов Д.А., Пушкин В.Н. Мышление и автоматы. - М.: Сов. радио, 1972. - 224 с.
13. Смирнов С.С., Болотова Л.С., Смолянинова В.А. Организация логического вывода на понятийно-объектной модели предметной области в системах поддержки принятия решений // Теоретические вопросы вычислительной техники и программного обеспечения: Межвуз. сб. научн. тр. - М.: МИРЭА, 2003. - С. 123-133.
14. Старых В.А., Болотова Л.С., Смолянинова В.А., Больных А.А. Обеспечение качества принятия решений в информационных системах с управлением: Труды XIV Всероссийской науч.-метод. конф. «Телематика 2007». - Т. 2. - СПб.: Университетские телекоммуникации, 2007. - С. 313-315. - ISBN 5-7577-0329-6.
15. Шрейдер Ю.А. Семиотические основы информатики. - М.: ИПКИР, 1975. - 80 с.
16. Шрейдер Ю.А., Шаров А.А. Системы и модели. - М.: Радио и связь, 1982. - 152
17. Gruber T.R. A translation approach to portable ontology specifications.// Knowledge acquisition Journal, 1993.v. 5, pp 199-220.
18. Guarino N. Formal Ontology and Information Systems. Proceeding of FOIS'98. Trento, Italy, Amsterdam: IOS Press, 1998, pp. 3-15
19. Guarino N. Understanding, building, and using Ontologies. A commentary to «Using explicit ontologies in KBS development» (by van Heijst, Schreiber and Wielinga)// International Journal of Human and Computer Studies, 1997, v. 46, № 2/3. pp. 293-310.
20. Stumme G., Medche A. FCA-Merge: Bottom-up merging of ontologies // 7th Int. Conf. On Artificial Intelligence, (IJCAI'01), Seattle, WA, 2001, P.225–230.