

ПРИМЕНЕНИЕ ОНТОЛОГИЙ

Онтологии



Онтология — описание на некотором формальном языке понятий некоторой предметной области и отношений между ними



На сегодняшний день в неоднородных ИС накапливается значительный объем знаний. И при интеграции таких систем возникает проблема систематизации и структурного представления знаний о разных предметных областях. Для решения данной проблемы можно использовать онтологические модели с целью получения формальной спецификации концептуализации.

Онтология применяется в системах управления знаниями с целью формального описания моделируемой части мира в виде словаря, разделяемого специалистами в выбранной предметной области. На основе этого общего словаря можно интегрировать различные источники знаний. Таким образом, используя общий словарь, можно понимать и сравнивать различные информационные системы

Онтологию определяют как спецификацию концептуализации предметной области и как средство представления семантики информационных единиц. Онтологию предметной области можно использовать для описания информационных объектов, их свойств и связей. Информация об этих объектах может храниться в разных источниках, и чтобы увидеть полную картину, необходимо построить информационную модель

КЛАССИФИКАЦИЯ ОНТОЛОГИЙ

- ▣ **Онтологии верхнего уровня.** В них содержатся общие знания для нескольких предметных областей, описываются наиболее общие концепты (время, событие, действие и т. д.)
- ▣ **Онтологии, ориентированные на предметную область.** По назначению аналогичны с онтологиями верхнего уровня, но область интереса ограничена одной предметной областью. Могут использовать специализацию терминов, расположенных в онтологиях верхнего уровня. Используются экспертами по предметным областям для аннотирования информации.
- ▣ **Онтологии, ориентированные на задачу.** Это онтологии, используемые конкретными прикладными программами и содержащие термины, которые применяются при разработке ИС, выполняющих конкретные задачи.

МЕТОДОЛОГИЯ ПОСТРОЕНИЯ ОНТОЛОГИИ ПРЕДПОЛАГАЕТ РЕШЕНИЕ СЛЕДУЮЩИХ ЗАДАЧ

- определить цель и область применения;
- построить онтологию с использованием специализированного языка представления знаний;
- добиться общего понимания структуры информации;
- обеспечить использование знаний в предметной области;

ПРИ СОЗДАНИИ ОНТОЛОГИИ НЕОБХОДИМО:

- 1) провести онтологический анализ. Составляется словарь терминов, включающий в себя описание характеристик объектов и процессов информационной системы, описываются логические взаимосвязи между понятиями предметной области;
- 2) выделить концепты – базовые понятия;
- 3) определить количество уровней абстракции;
- 4) распределить концепты по уровням;
- 5) построить связи между концептами — определить отношения и взаимосвязи с базовыми понятиями;
- 6) проконсультироваться с различными специалистами для исключения противоречий и неточностей.

ФОРМАЛЬНАЯ МОДЕЛЬ ОНТОЛОГИИ

Онтология – формальное явное описание понятий (концептов) в рассматриваемой предметной области, свойств и атрибутов каждого понятия (слотов), и ограничений, наложенных на слоты (фацетов). Слоты иногда называют ролями. Онтология вместе с набором индивидуальных экземпляров концептов образует базу знаний.

Формальная модель онтологии представляется в виде "тройки конечных множеств" $O = \langle T, R, F \rangle$, где:

- ▣ T – термины предметной области, которую описывает онтология O ;
- ▣ R – отношения между терминами заданной предметной области;
- ▣ F – функции интерпретации, заданные на терминах и/или отношениях онтологии O .

Под отношениями понимают тип взаимодействия между концептами («часть-целое», «является подклассом», «оказывает воздействие», «похоже на» и т. п.). Для моделирования утверждений используются аксиомы.

Protégé – свободно распространяемый редактор, предназначенный для проектирования онтологий предметных областей. Редактор строит онтологии с использованием языка OWL.

PROMPT является расширением системы *Protégé* и служит для объединения и группировки онтологий. *PROMPT* выводит пользователю список операций для объединения двух онтологий, а также список конфликтов и их возможные решения. Пользователь выбирает необходимое действие, список формируется снова пока не будет готова новая онтология.

Такие инструментальные средства используются как для проектирования, так и для анализа онтологий, выполняя типовые операции, например:

- ▣ **выравнивание** – нахождение и установление соответствий в обеих онтологиях;
- ▣ **отображение** – нахождение семантических зависимостей между элементами разных онтологий;
- ▣ **объединение** – создание результирующей онтологии по двум другим.

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ИНТЕГРАЦИИ ДАННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ (ИС)

Как правило, объектная схема ИС включает в себя элементы, которые соответствуют сущностям разных предметных областей, каждый объект характеризуется значениями набора атрибутов и представляется как множество упорядоченных пар вида

$$u = \langle a_i, d_i \rangle, (1)$$

где a_i – атрибут объекта; d_i – значение атрибута $i \in [1...n]$; n – количество атрибутов.

Базовым понятием модели является концепт (класс объектов). Каждый концепт онтологии информационной системы определяется как единица знания и идентифицируется по имени и характеризуется типом. Поэтому концепт задается как

$$C_i = (Name_i, type_i), (2)$$

где $Name_i$ – уникальное имя (идентификатор) i -го концепта; $type_i$ – тип i -го концепта (абстрактный, представимый, либо составной).

Абстрактный тип - список, массив и т.д. Представимый тип – числа, строки, изображения и т.д. Составной тип – агрегация неоднородных или однородных конструкций (понятие, атрибут, отношение).

Пример, в котором определяются понятия с абстрактными типами (*Помещение, ПоточнаяЛекционнаяАудитория, ВместимостьПомещения*).

```
{ОнтологияАудиторныйФонд;
in: module;
kind: ontology;
type:
{ Помещение;
in: type, concept;
ВместимостьПомещения: ВместимостьПомещения;
metaslot inverse: ВместимостьПомещения.ИзПомещение.
end
},
{ ПоточнаяЛекционнаяАудитория;
in: type, concept;
supertype: Помещение;
ПлощадьПомещения: {in: predicate, invariant;
{ predicative; {
all a/ ПоточнаяЛекционнаяАудитория
(a. ВместимостьПомещения > 50)
}}}
},
{ ВместимостьПомещения;
in: type, concept;
ИзПомещение: Помещение;
metaslot inverse: Помещение.ВместимостьПомещения;
end
};
}
```

Ниже приведен пример множества терминов
концепта *Аудитория*.

```
{ Аудитория;  
in: metaclass;  
Term_section:  
  {Комната;  
  Помещение;  
  Конференц-зал;  
  Сцена;  
  Трибуна;  
  Доска } }.
```

АЛГОРИТМ ИНТЕГРАЦИИ ИНФОРМАЦИОННЫХ СИСТЕМ НА ОСНОВЕ ОНТОЛОГИЙ

Интеграция концептов исходных онтологий заключается в размещении данного концепта интегрируемой онтологии в иерархию базовой онтологии. Общая схема работы алгоритма состоит во вставке интегрируемого класса концептов в базовую иерархию на максимально низкий уровень. Проход вниз по иерархии реализован рекурсивно.

1. Сравнение онтологий.

Выполняется выборка исходных онтологии O и O' интегрируемых ИС. Предполагается, что O является базовой, основной онтологией, а O' - это интегрируемая онтология. Рассчитываются весовые коэффициенты для мер семантической близости концептов, а также пороговые значения для классификации отношений между концептами C .

2. Интеграция концептов.

Шаг 1. Формируется множество $C1'$ подчиненных концептов с корневой вершиной $C1$ базовой онтологии O и множество $C2'$ подчиненных концептов с корневой вершины $C2$ интегрируемой онтологии O' .

Шаг 2. Начало цикла. В цикле осуществляется сопоставление и интеграция элементов множества $C2'$ с элементами множества $C1'$, т. е. концепт из множества $C2'$ интегрируется с концептом-вершиной $C1$ в иерархии онтологии.

Шаг 3. Рассчитывается мера семантической близости для концептов из множества $C1'$ и концептов из множества $C2'$.

Шаг 4. В соответствии с пороговыми значениями меры близости определяется вид семантической зависимости между концептами и либо устанавливается отображение, либо выполняется алгоритм разрешения конфликтов. Цикл выполняется, пока не будут проанализированы все элементы множества $C2$.

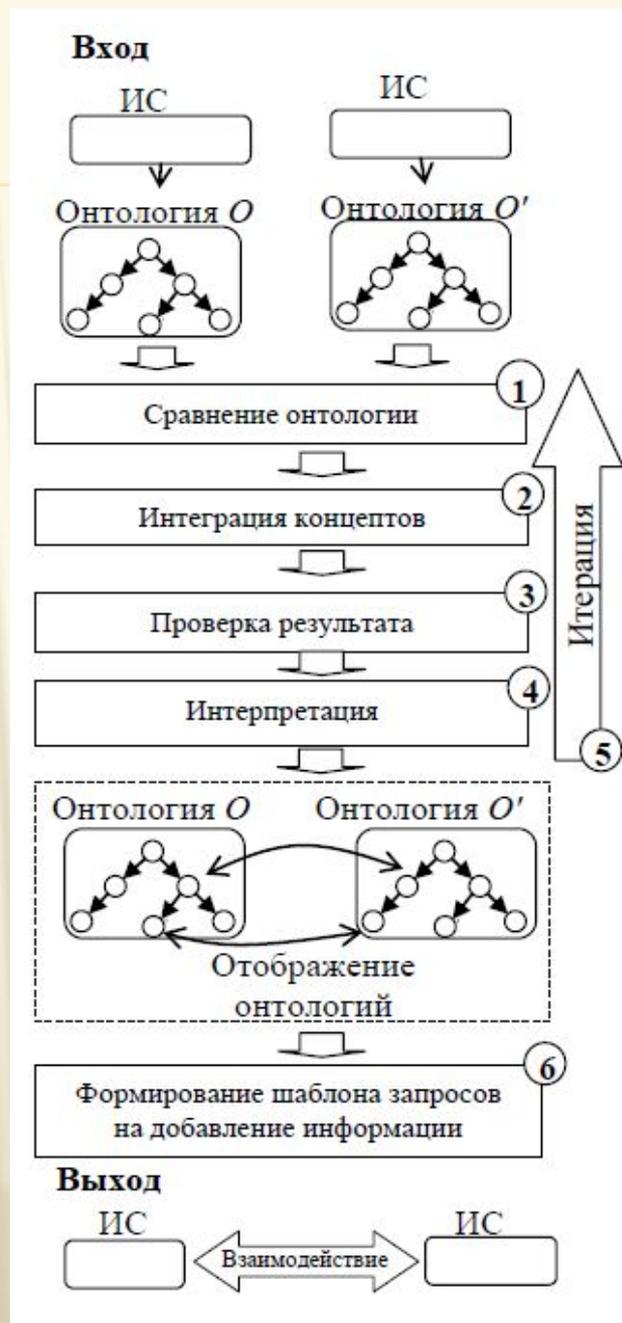
Если найдены зависимости z^1, z^5 , то отображение концептов устанавливается автоматически. Если найдены зависимости z^2, z^3, z^4 , то корректность построенного отображения подтверждается вручную.

3. Проверка результата. Проверка правильности составленной результирующей онтологии.

4. Интерпретация. Вывод результирующих отображений между концептами и атрибутами онтологий ИС.

5. Итерация. Повторение некоторых шагов алгоритма.

6. Установление отображений между элементами объектных схем информационных систем на основе связи онтологических концептов. После этого становится возможным генерация запросов на добавление информации из одной информационной системы в другую.



На первом этапе алгоритма учитываются варианты взаимного отношения концептов. Результат операции их сравнения приводит к пяти различным операциям над концептами.

1. Если **концепты эквивалентны**, то они представляют одно и то же понятие в онтологии, следовательно, должны быть «склеены» в один.

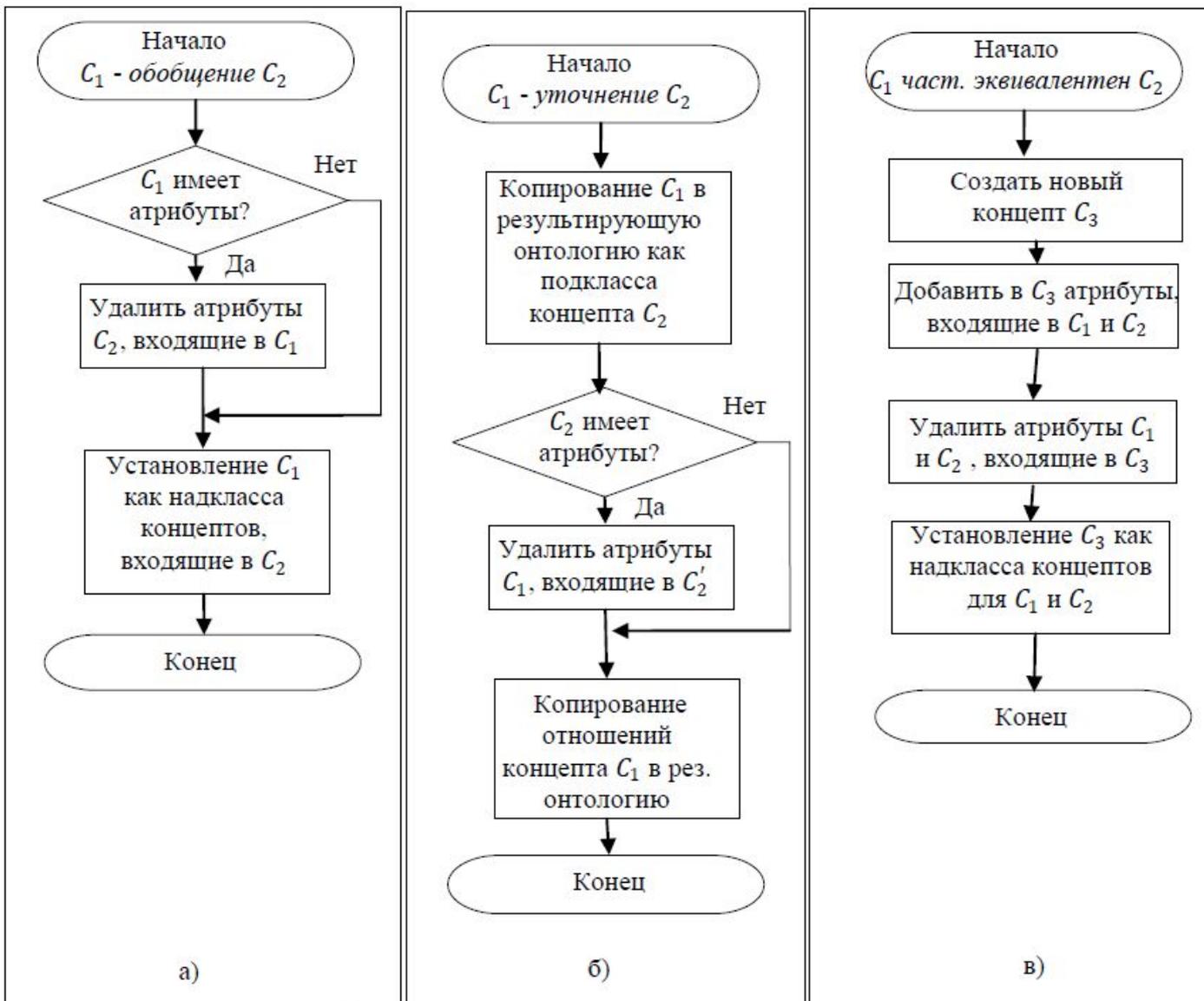
2. Если **концепт одной онтологии является обобщением соответствующего концепта другой онтологии**, такие концепты должны представляться как концепт и подкласс соответственно, причем совпадающие атрибуты должны быть удалены из подкласса, так как они будут унаследованы от надкласса (т.к. отношение «класс-подкласс» является отношением частичного порядка).

3. Если **концепт одной онтологии является уточнением соответствующего концепта другой онтологии**, такие концепты должны представляться как подкласс и класс соответственно, причем совпадающие атрибуты должны быть удалены из подкласса, так как они будут унаследованы от надкласса. Здесь необходимо учитывать все существующие отношения этих концептов.

4. Если **концепты двух онтологий частично эквивалентны**, то они представляют собой схожие понятия, то есть должны иметь общий надкласс, являющийся их обобщением (заметим, этот надкласс не присутствовал ни в одной из исходных онтологий), при этом совпадающие атрибуты должны быть удалены из подкласса, так как они будут унаследованы от обобщающего концепта.

5. **В одной онтологии может отсутствовать эквивалентный ему концепт из другой онтологии.** Ситуация, когда мера близости концептов меньше порога, при котором можно считать, что концепт в исходной онтологии отсутствует. В данном случае вычисляются меры семантической близости между данным концептом и всеми концептами исходной онтологии. Выбираются концепты, у которых мера близости S максимальна и установлено отображение в результирующую онтологию. Если условие выполняется, тогда концепт копируется в результирующую онтологию с атрибутами и связями как подкласс концепта, у которого существует отображение.

После построения результирующей онтологии ИС становится возможным интерпретировать информацию из одной ИС средствами другой ИС.



Блок-схема алгоритма взаимного позиционирования концептов: а) обобщение; б) уточнения; в) неполного соответствия

Онтологии предметной области и задачи

- Функциональная конфигурация (ФК) полностью соответствует определению онтологии предметной области.
- Она представляет собой формализованное описание объектов определенного типа и знаний относительно их свойств.
- Множество ОК представляется в виде И-ИЛИ графа, определяющего структуру конфигурации изделия

