



СПБГЭТУ «ЛЭТИ» ПЕРВЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ



3.3 Фреймы, как модель представления знаний. Управление выводом во фреймовых системах.

Семантические сети. Вывод на семантических сетях

Фреймы, как модель представления знаний.

Управление выводом во фреймовых системах.

Фрейм (от англ. Frame - рамка, несущая конструкция) - структура для описания стереотипной ситуации, содержащая множество характеристик ситуации, называемых *слотами*, и их значений.

Знания о мире могут быть представлены в виде фрейм-подобных структур. При анализе ситуации люди, как правило, используют свой опыт и адаптируют эти знания применительно к новой ситуации. Например, человек однажды остановившийся в гостинице, имеет представление о всех гостиничных номерах. Они содержат кровать, шкаф, ванную и т. д. Детали каждого номера (цвет портьер, расположение и тип выключателей и т. п.) могут отличаться. С фреймом гостиничного номера связана также информация, принимаемая по умолчанию. Например «Если нет простыней - нужно вызвать горничную», «если нужен лед - необходимо посмотреть в

СПБГЭТУ «ЛЭТИ», 2022 г.





холле», и т. п. Не надо иметь знания каждого нового гостиничного номера. Все элементы обобщенного номера организуются в концептуальную структуру, к которой человек обращается, когда останавливается в гостинице.

Эти высокоуровневые структуры можно представить в семантической сети, организуя ее как совокупность отдельных сетей, каждая из которых представляет некоторую стереотипную ситуацию. Фреймы так же, как *объектно-ориентированные системы*, обеспечивают механизм подобной организации, представляя сущности, как структурированные объекты с поименованными ячейками и связанными с ними значениями. Таким образом, фрейм или схема рассматривается как единая сложная сущность.

Общий вид фреймовой структуры:

(Имя_фрейма, (Имя_слота_1, Значение_слота_1),
(Имя_слота_2, Значение_слота_2),
... ..
(Имя_слота_N, Значение_слота_N)
)

Например, гостиничный номер и его компоненты могут быть описаны рядом отдельных фреймов:

Помимо кровати во фрейме должен быть представлен:

- стул: ожидаемая высота – от 20 до 40 см, число ножек – 4, значение по умолчанию – предназначен для сидения.
- гостиничный телефон: это вариант обычного телефона, но расчет за переговоры связывается с оплатой гостиничного номера. По умолчанию существует специальный гостиничный коммутатор, и человек может использовать этот телефон для заказа еды в номер, внешних звонков и получения других услуг.

На рис. 11.1 изображен фрейм, представляющий гостиничный номер:





Рис. 11.1 - Фрейм гостиничного номера.

Как видно из примера, фрейм может содержать любое число слотов, в конкретных системах некоторое число слотов м.б. обязательными, другие - необязательными.

Подробная структура фрейма:

(Имя_фрейма, (Имя_слота_1,

Указатель_наследования,

Указатель_типа_данных,

Значение_слота,

Демон

),

... ..

(Имя_слота_N,

Указатель_наследования,

Указатель_типа_данных,

Значение_слота,



Демон

),

)

Имя_фрейма - идентификатор, уникальный в данной фреймовой системе

Имя_слота_X - уникальный идентификатор слота в пределах данного фрейма

Указатель_наследования - используется во фреймовых системах иерархического типа, позволяющих организовать иерархию знаний. Показывает, какую информацию об атрибутах слотов в фрейме верхнего уровня наследуют слоты с теми же именами нижнего уровня.

В сети каждое понятие представляется узлами и связями на одном и том же уровне детализации. Однако очень часто для одних целей объект необходимо рассматривать как единую сущность, а для других – учитывать детали его внутренней структуры. Например, обычно нас не интересует механическое устройство автомобиля, пока что-то не сломается. При обнаружении поломки мы достаем схему автомобильного двигателя и пытаемся устранить проблему.

Системы фреймов поддерживают наследование классов. Значения ячеек и используемые по умолчанию значения класса наследуются через иерархию класс/подкласс и класс/член. Например, гостиничный телефон можно реализовать как подкласс обычного телефона, обладающий следующими особенностями: внешние звонки проходят через коммутатор гостиницы (для расчета), и гостиничные услуги можно заказывать по прямому номеру.

Типовые указатели наследования:

U - unique, «Уникальный», указывает, что каждый фрейм может иметь слоты с различными, уникальными значениями.

Фрейм высшего уровня





Фрейм	Слот	Значение
Человек	Масса	50

Фрейм среднего уровня

Фрейм	Слот	Значение
Ребенок	Масса	25

Фрейм низшего уровня

Фрейм	Слот	Значение
Петя	Масса	32

Как видно в примере, каждый слот имеет свое собственное независимое значение.

S - same, «такой же», - все слоты должны иметь одинаковые значения

Фрейм верхнего уровня

Фрейм	Слот	Значение
Человек	Масса	50

Фрейм среднего уровня

Фрейм	Слот	Значение
Ребенок	Масса	50

Фрейм низшего уровня

Фрейм	Слот	Значение
Петя	Масса	50

Тут все фреймы в иерархии имеют одинаковое значение.

R - range - значение слотов фрейма должны находится в диапазоне указанных значений слотов фрейма верхнего уровня

Фрейм верхнего уровня





Фрейм	Слот	Значение
Человек	Масса	30 -- 150

Фрейм среднего уровня

Фрейм	Слот	Значение
Ребенок	Масса	10 -- 40

Фрейм верхнего уровня

Фрейм	Слот	Значение
Петя	Масса	32

Значение слота последующего фрейма находится в диапазоне, определенном «родителем»

O - override, «Игнорировать» - при отсутствии указания, значение слота верхнего уровня наследуется слотом нижнего уровня, а в случае явного определения нового значения наследование не используется.

1 - Фрейм высшего уровня

Фрейм	Слот	Значение
Человек	Масса	70

2 - Фрейм среднего уровня

Фрейм	Слот	Значение
Ребенок	Масса	50

3 - Фрейм низшего уровня

Фрейм	Слот	Значение
Петя	Масса	50

Во втором фрейме значение слота было переопределено, а в третьем - осталось по умолчанию.

Указатель_типа_данных: аналог типизации переменных в программировании.

Возможные типы данных:





- FRAME - указатель на другой фрейм
- INT
- REAL
- BOOL
- TEXT
- LIST
- TABLE
- EXPR

Значение_слота - тип значения должен совпадать с явно указанным.

Демон - процедура, автоматически запускаемая при выполнении некоторого условия.

Демоны запускаются при обращении к соответствующему слоту и выполняют определенные действия со значением слота.

Типичные демоны	
IF-NEEDED	Запускается, если в момент обращения к слоту его значение не установлено
IF-ADDED	запускается при записи в слот некоторого значения
IF-REMOVED	запускается при удалении значения слота

Демоны, по сути, являются разновидностями *присоединенных процедур* - программ, которые используются для определения значения слота. Например, при каждом изменении определенной ячейки в системе можно запускать процедуру контроля соответствия типов или тест непротиворечивости.

Подобные процедурные вложения являются важным свойством фреймов, так как они позволяют связать фрагменты программного кода с соответствующими сущностями фреймового представления. Например, в





базу знаний можно включить возможность генерировать графические образы.

Присоединенные процедуры запускаются по сообщениям, переданным из другого фрейма.

Демоны и присоединенные процедуры представляют собой *процедурные знания* в составе фреймов. Остальные знания в данной системе - декларативные.

Фреймы расширяют возможности семантических сетей, позволяя представлять сложные объекты не в виде большой семантической структуры, а в виде единой сущности (фрейма). Это также позволяет естественным образом представить стереотипные сущности, классы, наследование и значения по умолчанию.

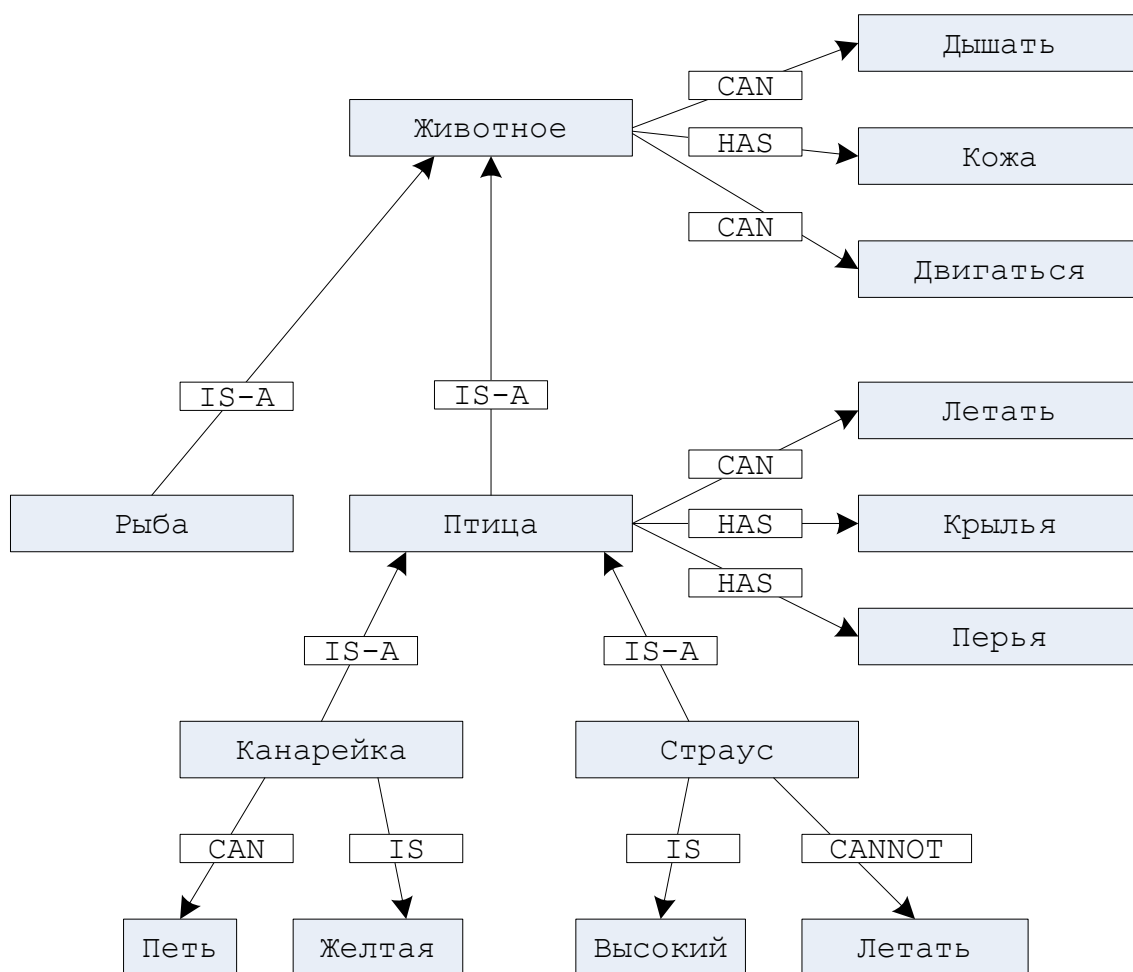
Семантические сети. Вывод на семантических сетях.

Формально, *семантическая сеть* (СС) представляет собой помеченный ориентированный граф. Вершины СС соответствуют сущностям предметной области - объектам, событиям, свойствам, процессам, явлениям.





Семантическая сеть, разработанная для исследования по хранению информации



Системы наследования позволяют запоминать информацию на самом высоком уровне абстракции, что уменьшает размер баз знаний и помогает избежать противоречия. Наследование также помогает поддерживать непротиворечивость базы знаний при добавлении новых классов и объектов.

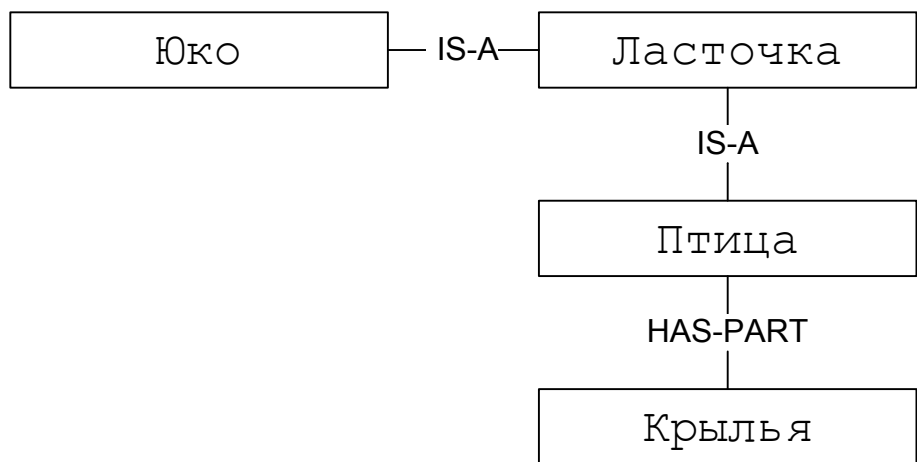
С точки зрения математической логики, каждая вершина соответствует элементу предметного множества, а дуга – предикату.





Классификация семантических сетей		
По сложности структуры		<ul style="list-style-type: none">• Простые сети - вершины не имеют внутренней структуры• Иерархические сети
По количеству типов отношений		<ul style="list-style-type: none">• Однородные - только один тип отношений• Неоднородные
По арности (количеству понятий, связанных одной дугой)		<ul style="list-style-type: none">• Бинарные• N-арные

Пример простой СС:



На языке логики предикатов это будет выглядеть следующим образом:

$$\forall x [P(x) \rightarrow Q(x)]$$

где: P(x) - "x - ласточка"

Q(x)- "x - птица"

Наиболее часто возникает потребность в описании отношений между элементами, множествами и частями объектов.

Отношение между объектом и множеством, обозначающим, что объект принадлежит этому множеству, называется отношением классификации (IS-A). Говорят, что множество (класс) классифицирует свои экземпляры. Название произошло от английского «IS-A». Иногда это отношение именуют также





MemberOf. Отношение IS-A предполагает, что свойства объекта *наследуются* от множества.

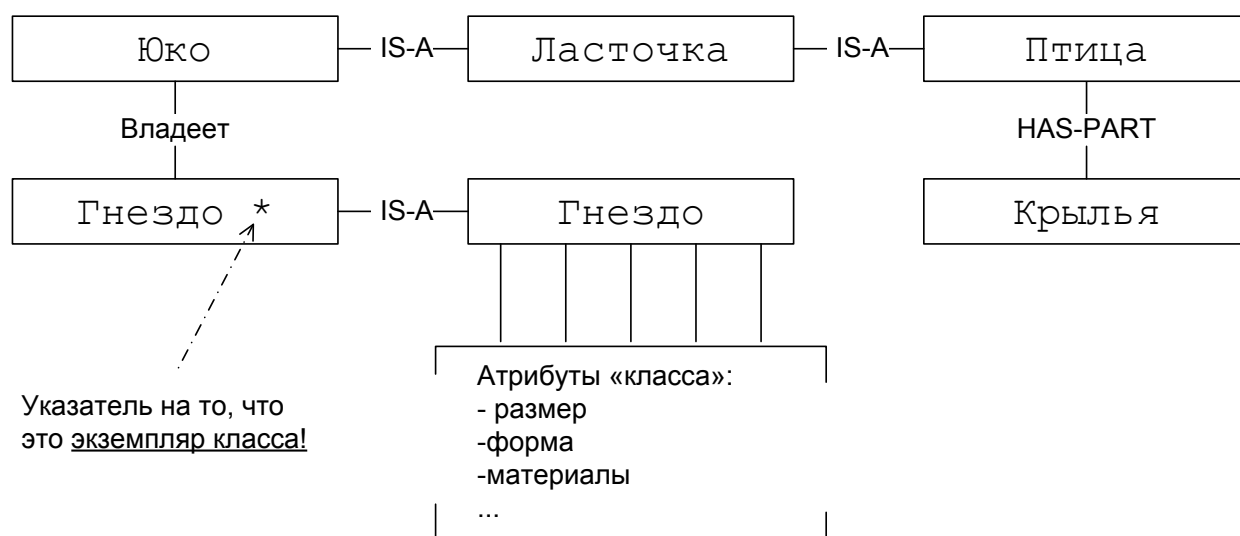
Обратное к IS-A отношение используется для обозначения примеров, поэтому часто называется «Example».

Отношение между надмножеством и подмножеством называется АКО — «A KIND OF» («разновидность»). Альтернативные названия — «SubsetOf» и «Подмножество». Это отношение определяет, что каждый элемент первого множества входит и во второе (выполняется IS-A для каждого элемента), а также логическую связь между самими подмножествами: что первое не больше второго и свойства первого *множества* наследуются вторым.

Объект, как правило, состоит из нескольких частей, или элементов. Например, компьютер состоит из системного блока, монитора, клавиатуры, мыши и т. д. Важным отношением является HAS-PART, описывающее части/целые объекты.

В С-сети имеет место иерархия понятий, позволяющая делать выводы на основе наследования.

Наследование в данном случае обусловлено транзитивностью отношений IS-A и HAS-PART:





Часто отношения рассматриваются как сущности, имеющие собственные атрибуты, например отношение владения может характеризоваться периодом владения.



Использование данной сети можно реализовать процедурой знаний, отвечающей на различные вопросы: Чем владеет Юко? В течении какого периода?

Вывод на семантических сетях

В основе процедур вывода на семантической сети лежит сопоставление частей сетевой структуры (подграф).

При этом строится подсеть, соответствующая запросу.

Пример: Запросу «Чем владеет Юко?» соответствует подсеть:

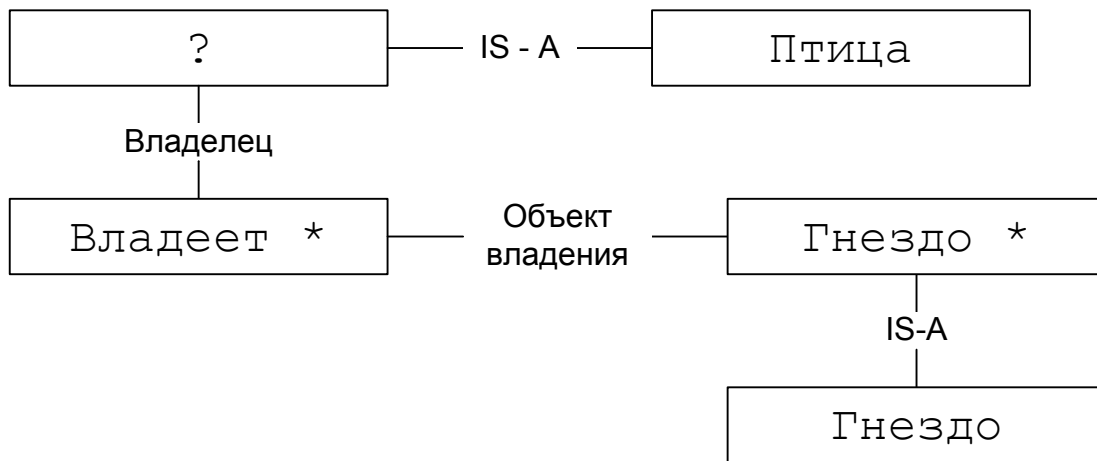




При обработке запроса проводится сопоставление графа с базой знаний (семантической сетью).

Отыскивается вершина [владеет *] (или дуга «владелец», направленная к вершине [Юко]). Затем определяется вершина, к которой ведет дуга «объект владения». Метка вершины - ответ на вопрос.

Запросу «Существует ли птица, которая владеет гнездом?» соответствует следующий граф.



Теория семантических сетей получает применение при создании *семантической паутины* - метода представления данных в сети Интернет, при котором возможна их семантическая обработка.

Семантическая паутина – часть концепции развития Всемирной паутины и сети Интернет, принятой консорциумом W3. Целью внедрения этой концепции является создание сети документов, содержащих





метаданные, существующей параллельно с ресурсами, предназначенными для восприятия человеком (в данном контексте *ресурсом* является любой объект, доступный в Интернете: веб-страница, изображение, аудио- или видеоматериал, и т. п.), и позволяющей машинам (программным *интеллектуальным агентам*) автоматически производить чёткие (недвусмысленные) заключения о свойствах этих ресурсов.

Техническую часть семантической паутины составляет семейство стандартов на языки описания, включающее XML, XML Schema, RDF, RDF Schema, OWL, а также некоторые другие.

Достоинства и недостатки семантических сетей

- **Достоинства:** Подобная форма - самая удобно воспринимаемая человеком.
- **Недостатки:** Трудности при построении сложных сетей и попытке учета особенностей естественного языка.

