

## **Практическое задание №7**

### **Представление знаний в виде семантические сети**

#### **1.1 Цель работы**

Изучение основных правил представления знаний в экспертных системах (ЭС) в виде семантических сетей

#### **1.2 Теоретические сведения**

**Семантическая сеть** - структура для представления знаний в виде узлов, соединенных дугами. Самые первые семантические сети были разработаны в качестве языка-посредника для систем машинного перевода, а многие современные версии до сих пор сходны по своим характеристикам с естественным языком. Однако последние версии семантических сетей стали более мощными и гибкими и составляют конкуренцию фреймовым системам, логическому программированию и другим языкам представления. Начиная с конца 50-ых годов были создано и применены на практике десятки вариантов семантических сетей. Несмотря на то, что терминология и их структура различаются, существуют сходства, присущие практически всем семантическим сетям:

- узлы семантических сетей представляют собой концепты предметов, событий, состояний;
- различные узлы одного концепта относятся к различным значениям, если они не помечено, что они относятся к одному концепту;
- дуги семантических сетей создают отношения между узлами-концептами (пометки над дугами указывают на тип отношения);
- некоторые отношения между концептами представляют собой лингвистические падежи, такие как агент, объект, реципиент и инструмент (другие означают временные, пространственные, логические отношения и отношения между отдельными предложениями);
- концепты организованы по уровням в соответствии со степенью обобщенности так как, например, сущность, живое существо, животное, плотоядное;

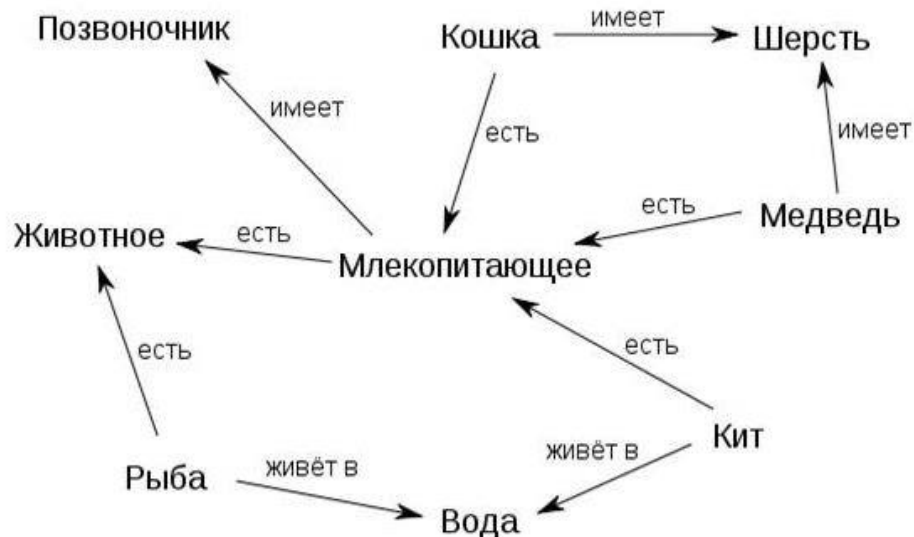


Рис 1 - Пример семантической сети

Для всех семантических сетей справедливо разделение по арности и количеству типов отношений.

По количеству типов отношений, сети могут быть однородными и неоднородными. Однородные сети обладают только одним типом отношений (стрелок), например, таковой является вышеупомянутая классификация биологических видов (с единственным отношением АКО). По арности (бинарные, N-арные): типичными являются сети с бинарными отношениями (связывающими ровно два понятия). Бинарные отношения очень просты и удобно изображаются на графе в виде стрелки между двух концептов. Кроме того, они играют исключительную роль в математике; на практике могут понадобиться отношения, связывающие более двух объектов — N-арные. При этом возникает сложность — как изобразить подобную связь на графе, чтобы не запутаться. Концептуальные графы снимают это затруднение, представляя каждое отношение в виде отдельного узла.

В поисковых системах используются два понятия, связанные с установлением семантических связей: тезарус и отнология. Необходимы для лучшего поиска содержимого текста.

Важную роль в анализе и формировании формализованного представления текстовых документов играют специальные словари, называемые тезаурусами.

*Тезаурус* — это словарь основных понятий языка, обозначаемых отдельными словами или словосочетаниями, с определенными семантическими связями между ними. Существует два способа создания тезаурусов — ручной и автоматический.

Тезаурус, созданный вручную, может быть универсальным, независимым от конкретной коллекции документов. Автоматическое создание тезаурусов осуществляется на основе заданных коллекций текстовых документов, поэтому такие тезаурусы предназначены для работы именно с этими коллекциями.

### *Онтология*

Для адекватной интерпретации пользователем и/или системой содержания текстовых документов и поисковых запросов, представленных на естественном языке, система должна располагать контекстом, который определяет основные понятия предметной области и различного рода семантические связи между ними. Это описание не зависит от содержащихся в системных коллекциях конкретных документов и представляет собой спецификацию концептуализации предметной области системы. Выраженная явным образом такая спецификация называется онтологией предметной области. Онтологии получили в последние годы широкое распространение в разнообразных информационных системах, основанных на знаниях, в инженерии" знаний, в решении проблем семантической интеграции информационных ресурсов и т.д.

#### **1.2.1 Примеры семантических информационных сетей и систем**

Группа учёных из университета Пенсильвании (Pennsylvania State University) объявила о регистрации патента на «семантическую» систему визуального поиска ALIPR (Automatic Linguistic Indexing of Pictures in Real-Time). Суть нового сервиса в объединении «пиксельного» и смыслового анализа изображений. В базе ALIPR находится более миллиона изображений, к каждому из которых проставлены теги – группа смысловых «ключей». Основная идея в том, что восприятие тех или иных объектов у пользователей Интернета всё же подчиняется законам нормального распределения, и, следовательно, наиболее популярные связки «понятие – объект» должны совпасть с ожиданиями. В настоящий момент система оперирует 332 понятиями, но это не означает, что найти можно картинки лишь по столь ограниченному набору слов.

WordNet — это семантическая сеть для английского языка, разработанная в Принстонском университете, и выпущенная вместе с сопутствующим программным обеспечением под некомпилефтной свободной лицензией.

Freebase – семантическая база данных. Freebase работает не только через текстовый поиск, а что наиболее важно, и через — MQL (Metaweb Query

Language). MQL схоже с JSON, текстовый формБаза знаний пользовательского интерфейса включает в себя следующую информацию:

- описание команд пользовательского интерфейса – описываются классы команд пользовательского интерфейса, которые служат для инициирования пользователем каких-либо действий в системе.
- описание структуры главного меню;
- описание внешних языков представления знаний;
- описание используемых компонентов – используются интерфейсом для организации диалога с пользователем. К примеру, при установке содержимого sc-рамки пользователю отображается список редакторов, которые имеются в базе знаний;
- протокол пользовательских действий – строится во время работы системы. Он необходим для построения портрета пользователя, а также обеспечения отмены пользовательских действий;
- портрет пользователя – строится на основании анализа действий пользователя. Включает в себя всю необходимую информацию о пользователе. В качестве такой информации может выступать предпочитаемая стилистика размещения, наиболее часто используемые внешние языки;ат обмена данными), только с более широкими возможностями. С его помощью можно составить любой запрос в Freebase и ответом будет тот же запрос, но уже со вставленными результатами поиска.

Семантическая технология проектирования баз знаний, программ, машин обработки знаний, пользовательских интерфейсов, интеллектуальных справочных систем, интеллектуальных обучающих систем

Машина обработки знаний пользовательского интерфейса имеет свою специфику – кроме обычных операций обработки знаний в ней присутствуют “эффекторные” и “рецепторные” операции. “Эффекторные” операции осуществляют вывод информации пользователю, реагируя на событие в памяти системы. “Рецепторные” операции реагируют на события пользователя (нажатие клавиши, перемещение мыши) и изменяют состояние памяти системы. В машину обработки знаний пользовательского интерфейса обязательно входят следующие операции:

1 Операции работы с командами:

- добавление дочерней команды в неатомарную команду;
- удаление дочерней команды из неатомарной команды;

2 Операции работы с компонентами:

- добавление компонента;
- удаление компонента;

### 3 Операции работы с окнами:

- создание окна указанного типа;
- удаление окна;
- открытие/закрытие окна;

### 4 “Эффекторные” операции:

- инициирование вывода ответа на вопрос заданный пользователем;
- вывод информационной конструкции пользователю;

5 Рецепторные” операции (список может быть расширен в зависимости от используемых внешних устройств):

- нажатие кнопки на клавиатуре;
- нажатие клавиши мыши;
- вращение колеса прокрутки мыши;

В состав предлагаемой семантической технологии компонентного проектирования пользовательских интерфейсов интеллектуальных систем входит:

- унифицированная семантическая модель пользовательских интерфейсов интеллектуальных систем;
- библиотека совместимых ip-компонентов пользовательских интерфейсов;
- инструментальные средства компонентного проектирования пользовательских интерфейсов;
- методика компонентного проектирования пользовательских интерфейсов;
- методика обучения проектированию пользовательских интерфейсов;
- интеллектуальная help-система по семантической технологии компонентного проектирования пользовательских интерфейсов интеллектуальных систем.

Поставленные задачи соответствуют разработке выше перечисленных подсистем, которые входят в состав технологии.

*Интеллектуальная help-система*, по семантической технологии проектирования пользовательских интерфейсов интеллектуальных систем, решает задачу помощи разработчикам в процессе проектирования пользовательских интерфейсов. Данная интеллектуальная система строится с использованием семантической технологии проектирования интеллектуальных систем. Её база знаний содержит следующие разделы:

1 Унифицированная семантическая модель пользовательских интерфейсов интеллектуальных систем. В данном разделе содержится полное формальное описание модели пользовательских интерфейсов, это дает возможность пользователю (разработчику) значительно быстрее усвоить теорию и понять принципы лежащие в основе проектируемых им интерфейсов.

2 Библиотека совместимых ip-компонентов пользовательских интерфейсов интеллектуальных систем. В этом разделе приводится полное

формальное описание библиотеки ip-компонентов. Это позволяет разработчику значительно быстрее освоить принципы работы с библиотекой и тем самым сократить сроки проектирования.

3 Интегрированные средства проектирования пользовательских интерфейсов. Раздел базы знаний посвященный писанию средств, которые позволяют автоматизировать и интеллектуализировать процесс проектирования.

4 Методика проектирования пользовательских интерфейсов интеллектуальных систем. Содержит детальное описание процесса проектирования.

5 Методика проектирования пользовательских интерфейсов интеллектуальных систем. Содержит детальное описание процесса проектирования.

6 Методика обучения проектированию пользовательских интерфейсов интеллектуальных систем. Данный раздел базы знаний полностью посвящен процессу обучения проектированию. Он содержит информацию, которая должна помочь пользователю в освоении самой технологии.

7 Help-система по семантической технологии проектирования пользовательских интерфейсов интеллектуальных систем. Раздел, который призван помочь пользователю освоить саму help-систему.

Включение интеллектуальной help-системы в состав технологии позволяет не только сократить сроки проектирования, но и снизить требования, предъявляемые к начальной квалификации разработчика.

### **1.2.2 Интегрированные средства проектирования пользовательских интерфейсов**

Интегрированные средства проектирования пользовательских интерфейсов интеллектуальных систем решают задачу автоматизации и интеллектуализации процесса проектирования. В их состав входят инструментальные средства визуального проектирования, средства отладки и верификации проектируемых интерфейсов, библиотека ip-компонентов пользовательских интерфейсов и help-система. Интеграция указанных систем в рамках интегрированной среды, позволяет значительно ускорить процесс проектирования, а также повысить качество проектируемых интерфейсов.

В основе инструментальных средств визуального проектирования лежит пользовательский SCg-интерфейс. Это дает разработчику возможность проектирования пользовательских интерфейсов с использованием известной технологии WYSIWYG (What You See Is What You Get – «что вы видите то и получите», то есть результат проектирования выглядит так же, как и прототип во время разработки). Инструментальные средства верификации и отладки, пользовательских интерфейсов, представляют собой специализированную

интеллектуальную систему, которая позволяет находить и устранять ошибки в проектируемом пользовательском интерфейсе.

Эта система направлена на автоматизацию и последующую интеллектуализацию процесса верификации и отладки проектируемого интерфейса. Для нее выделено три уровня интеллектуализации:

- на первом уровне средства верификации и отладки позволяют осуществлять поиск ошибок в проектируемом интерфейсе;
- на втором уровне интеллектуализации добавляется возможность анализа и оптимизации проектируемого интерфейса;
- на третьем уровне интеллектуализации появляется возможность автоматического исправления ошибок.

Методика проектирования пользовательских интерфейсов является важной частью технологии, так как она описывает сам процесс проектирования. На первом этапе разработчику необходимо специфицировать проектируемый пользовательский интерфейс. Спецификация включает в себя список задач решаемых интерфейсом, описание внешних языков представления знаний. На этапе задачно-ориентированной декомпозиции специфицированный разработчиком интерфейс разбивается на интерфейсные подсистемы, которые могут разрабатываться параллельно. Это позволяет сократить сроки проектирования пользовательского интерфейса. Целесообразно проводить разбиение таким образом, чтобы максимальное количество подсистем уже имелось в библиотеке *ip*-компонентов пользовательских интерфейсов. После уточнения функциональных возможностей разрабатываемых подсистем производится их разработка с использованием семантической технологии проектирования интеллектуальных систем.

### **1.2.3 Установка и настройка компонентов ядра пользовательских интерфейсов**

Одной из сложностей при проектировании интерфейса является его платформенная ориентация. Чтобы решить данную проблему в библиотеку *ip*-компонентов входят компоненты, которые реализуют ядро пользовательских интерфейсов. Ядро пользовательских интерфейсов включает в себя весь набор необходимых интерфейсных операций, а также осуществляет интеграцию используемых *ip*-компонентов между собой. Таким образом, наличие ядра пользовательских интерфейсов в библиотеке *ip*-компонентов, позволяет разработчику не задумываться о погрузке разработанного им интерфейса на конкретную платформу. Стоит отметить, что разработанные как *sc*-системы *ip*-компоненты являются максимально независимыми от платформы, в отличие от инородных *ip*-компонентов.

### 1.2.4 Создание и настройка главного меню

Главным называется окно системы, в рамках которого осуществляется взаимодействие пользователя с системой, оно принадлежит к классу `sc.g-окон`. Кроме того оно является синглтоном (не может быть двух главных окон). Главное окно предоставляет возможность общения пользователя с системой, используя `SCg-интерфейс`. `SCg-интерфейс` – это целостная часть пользовательского интерфейса, обеспечивающая взаимодействие пользователя с интеллектуальной системой, с помощью `SCg-кода`. По такому же принципу называются и другие интерфейсы. К примеру: естественно языковой интерфейс (взаимодействие с помощью естественного языка).

В рамках главного окна, которое относится к классу `sc.g-окон`, отображаются элементы управления пользовательского интерфейса. `SC.g-узлы` управления пользовательским интерфейсом – `sc-узлы`, которые отображаются с помощью `SCg-кода` на экране, и имеет свою, отличную от других, отображаемых на экране `sc.g-узлов`, операционную семантику. Нажатие кнопки мыши на этих узлах приводит к инициированию некоторых действий. Такие `sc.g-узлы` могут быть резидентными и нерезидентными. Резидентные `sc.g-узлы` управления всегда присутствуют на экране в рамках главного окна, в то время, как нерезидентные узлы не всегда присутствуют на экране. При этом на экране могут присутствовать несколько синонимичных `sc.g-узлов`, которые изображают один и тот же `sc-узел`, но выглядят по-разному.

Выделяются следующие типы элементов управления:

1 Элементы управления, изображающие только объект действий. Они используются для отображения объектов находящихся в базе знаний, над которыми можно производить какие-либо действия. Примерами таких элементов управления служат пункты меню, отображающие разделы документации, открытые окна и так далее. По сути, пользователь не может инициировать с помощью них какую-либо команду. Он может лишь их использовать как аргументы для какой-либо команды;

2 Элементы управления, отображающие классы действий. По сути, представляют собой команды, для которых объект действия не определен. Нажатие клавиши мыши на эти элементы управления приводит к инициированию действия принадлежащего данному классу. К примеру, нажатие клавиши мыши на элемент управления отображающий класс команд “закрыть окно” приводит к инициированию команды, которая принадлежит данному классу и осуществляет закрытие указанного пользователем окна. Элементы управления, относящиеся к данному типу, отображаются на экране



в виде знаков атомарных классов пользовательских интерфейсных команд. Они могут отображаться в составе меню и отдельно. При отображении в составе меню, они появляются на экране при инициировании команды открытия пункта меню и таким образом являются не резидентными sc.g-узлами.

В свою очередь класс атомарных команд делится на следующие подклассы:

- команда-вопрос – команд, которая иницирует вопрос системе;
- команда редактирования – команда, которая иницирует операцию редактирования информационной конструкции (создание или удаление объектов, изменение типа объектов);
- команда просмотра – команда, которая иницирует действия связанные с информационных конструкций (навигация, масштабирование и т. д.).
- элементы управления, отображающие конкретные действия. Они отображают конкретные команды, для которых уже определен объект действия. Примером такого элемента управления является команда закрытия конкретного окна (в современных интерфейсах отображаемая в правом верхнем углу, в виде крестика).

### **1.2.5 Создание и использование ip-компонентов пользовательских интерфейсов**

Большое разнообразие пользовательских интерфейсов влечет за собой разработку большого числа ip-компонентов (под ip-компонентом понимается часть пользовательского интерфейса, которая может быть интегрирована и использована в различных системах). Стоит отметить, что ip-компоненты могут состоять из других ip-компонентов (уровень вложенности не ограничен). Таким образом, в качестве ip-компонентов, могут выступать уже проектированные пользовательские интерфейсы. Большое число ip-компонентов создает проблему их хранения и поиска. Для решения этой проблемы в технологию включена библиотека совместимых ip-компонентов пользовательских интерфейсов (далее библиотека ip-компонентов). Она представляет собой специализированную интеллектуальную систему, которая решает следующие задачи: хранение, поиск, добавление, удаление и сравнение ip-компонентов пользовательских интерфейсов. В рамках библиотеки, все ip-компоненты специфицируются и классифицируются. Их классификация производится по различным признакам. К примеру, в зависимости от решаемых задач ip-компоненты делятся на следующие классы:

- просмотрщик содержимого sc-ссылок;

- редактор содержимого sc-ссылок;
- транслятор содержимого sc-ссылки в SC-код;
- транслятор из SC-кода в содержимое sc-ссылки;
- транслятор базы знаний во внешнее представление.

Технология позволяет интегрировать в качестве ip-компонентов редакторы и просмотрщики, разработанные с использованием других технологий (далее инородные ip-компонентами пользовательского интерфейса). В основном они используются для просмотра и редактирования содержимого sc-ссылок. Это значительно позволяет сэкономить время на их разработке. База знаний библиотеки ip-компонентов включает в себя следующую информацию:

- спецификация ip-компонентов;
- история версий ip-компонентов;
- статистика использования ip-компонентов.

Машина обработки знаний состоит из следующего набора операций:

- поиск ip-компонента, по его спецификации;
- добавление нового ip-компонента;
- удаление ip-компонента;
- сравнение двух ip-компонентов

Пользовательский интерфейс библиотеки ip-компонентов строится на основе SCg-интерфейса (комплекс информационно-программных средств обеспечивающих общение интеллектуальных систем с пользователем на основе SCg-кода, как способа внешнего представления информации). Однако это не исключает возможность использования других способов диалога пользователя с библиотекой ip-компонентов. В рамках библиотеки ip-компонентов могут содержаться различные версии и модификации какого-либо ip-компонента. К примеру, ip-компонент просмотра sc.g-конструкций может иметь модификации, для отображения в которых может использоваться двухмерная, трехмерная или же многослойная визуализация, при этом каждая из модификаций компонента может иметь различные версии. Использование библиотеки ip-компонентов при проектировании пользовательского интерфейса прикладной системы позволяет значительно сократить сроки проектирования, а также снизить требования, предъявляемые к начальной квалификации разработчика. Это достигается за счет проектирования пользовательского интерфейса из уже заранее заготовленных модулей, что также позволяет повысить качество проектируемого интерфейса. При таком подходе основной проблемой является проблема интеграции компонент

между собой. Для её решения предлагается осуществлять взаимодействие между ip-компонентами через базу знаний.

Одной из сложностей при проектировании интерфейса является его платформенная ориентация. Чтобы решить данную проблему в библиотеку ip-компонентов входят компоненты, которые реализуют ядро пользовательских интерфейсов. Ядро пользовательских интерфейсов включает в себя весь набор необходимых интерфейсных операций, а также осуществляет интеграцию используемых ip-компонентов между собой. Наличие ядра пользовательских интерфейсов в библиотеке ip-компонентов, позволяет разработчику не задумываться о погрузке разработанного им интерфейса на конкретную платформу.

### **1.3 Задание**

Для предметной области сформировать базу знаний, соответствующую следующим требованиям:

- включить не менее 12 смежных понятий;
- для описания правил использовать, не менее 3 типов отношений.

Отчет о лабораторной работе должен содержать:

- перечисление понятий;
- перечень типов отношений, представленных в сети;
- графическое изображение семантической сети выбранной предметной области;
- представить два вида использования описания отношений между понятиями: интенциональная и экстенциональная сеть (желательно на выбранной предметной области).

*Варианты предметных областей:*

1 ЭС по выбору модели сотового телефона (компьютера, фотокамеры и т. д.) с необходимыми характеристиками.

2 ЭС определения психологического типа человека.

3 ЭС определения продолжительности жизни человека в зависимости от образа жизни.

4 ЭС определения растений (животных).

5 ЭС поиска неисправностей автомобиля и с выдачей рекомендаций по их устранению.

6 ЭС по кредитованию населения, предприятия и т.д.

### **1.4 Пример отчёта**

### 1.4.1 Пример 1

*Предметная область - Аэродинамика*

*Понятия:*

- 1) Самолет
- 2) Ту-134
- 3) Крылья
- 4) Мотор
- 5) Бензин
- 6) Летать
- 7) Законы аэродинамики
- 8) Птица
- 9) Орел
- 10) Клюв
- 11) Оперение

*Отношения:*

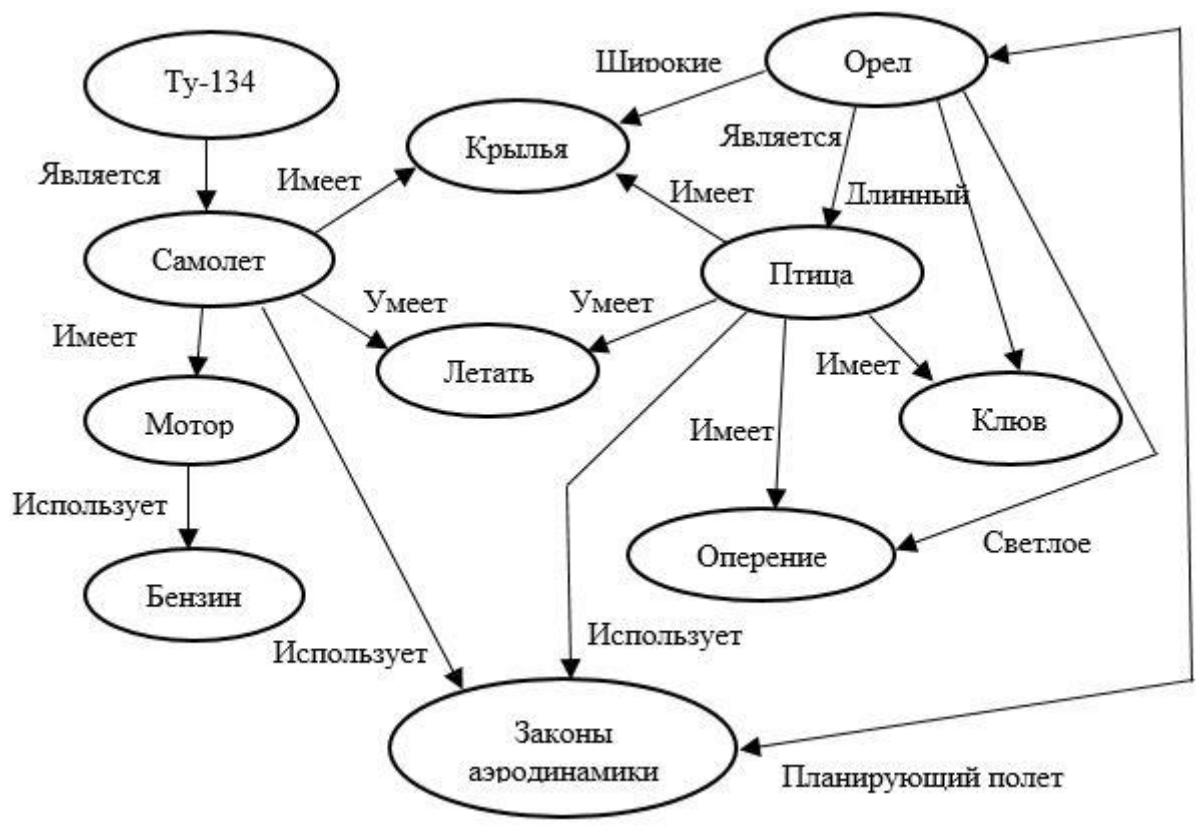
Тип "Множество-Подмножество"(a Kind of): Является

Тип "Функциональные связи": Имеет, Умеет, Использует

Тип "Атрибутивные": Широкие, Длинные, Светлое

Тип "Лингвистические": Планирующий полет

*Семантическая сеть:*



## 1.4.2 Пример 2

### Построение структурной схемы системы

*Цель работы:*

Освоить построение структурной схемы.

*Исходные данные:*

Система: Дорожный знак “кирпич”.

*Задачи работы:*

Построение структурной схемы системы дорожный знак ”кирпич”, с указанием компонентов этой структуры и их связей.

*Выполнение работы:*



*Вывод:*

В ходе выполнения данной лабораторной работы было освоено построение структурной схемы типа "Дорожный знак «въезд запрещен»".