

Задание на проектирование системы управления базой ПОНЯТИЙ

1. Общие требования

Система ЭЗОП (Элементов Задач и Определений) представляет собой программную среду для формализации знаний различных предметных областей. Она предназначена для формирования библиотеки формализованных понятий некоторой области знаний, а также для поддержки процесса формализации отдельных понятий и задач этой области знаний с использованием существующей библиотеки понятий. В системе должны быть предусмотрены средства тестирования и отлаживания формируемых понятий и библиотек, средства поддержки версий.

Основное отличие разрабатываемой системы от систем подобного рода, например системы Ontolingua (см. www.ksl.stanford.edu), состоит в возможности подстраивания языка системы к языку представляемой области знаний по мере пополнения библиотеки понятий понятиями и языковыми шаблонами этой области знаний.

Система ЭЗОП должна обеспечивать загрузку библиотек онтологий системы Ontolingua для работы с ними средствами системы ЭЗОП, а также перевод формализованных в системе ЭЗОП знаний в формат языка системы Ontolingua для обеспечения взаимодействия системы ЭЗОП с другими системами представления знаний.

Кроме того, в системе ЭЗОП должен быть обеспечен перевод задач, формализованных средствами ЭЗОП, на язык системы Mathematica для решения этих задач с использованием программных средств системы Mathematica.

Первоначальная версия системы должна работать на персональном компьютере под управлением Windows. Последующие версии должны обеспечивать удаленную многопользовательскую работу с системой в среде интернет.

2. Основные задачи и технология работы с системой

Основная функция системы – формирование формального описания понятия или задачи с использованием библиотеки уже формализованных понятий или фрагментов задач.

В каждый момент времени может быть только одно формируемое понятие, которое называется текущим. Текущее понятие можно дополнять новыми предложениями, редактировать и тестировать, задавая к миру текущего понятия вопросы или просматривая словари текущего понятия.

В процессе формализации понятия или задачи пользователю должен быть обеспечен доступ к словарям шаблонов, к текстам понятий в библиотеке понятий и текстам в архивах.

Текущее понятие можно поместить в библиотеку понятий с сохранением прежнего имени (но в новой версии) или под новым именем. Любое понятие из библиотеки понятий можно сделать текущим или использовать внутри другого понятия.

Система должна обеспечивать процессы просмотра и ведения библиотеки понятий, включая процессы пополнения библиотеки, организации разделов, сортировки, поиска понятий, уничтожения версий понятий при условии, если на них не ссылаются другие понятия.

С каждым понятием связывается имя понятия, внутренний идентификатор, текст понятия, языковые шаблоны и термины, введенные внутри данного понятия, а также внутреннее представление понятия, которое строится по тексту понятия.

Внутреннее представление понятия используется для ответов на вопросы и для контекстной интерпретации новых предложений и проверки их правильности. Внутреннее представление понятия (или задачи) строится из пустого и достраивается по каждому предложению понятия. При построении ответов на вопросы к текущему понятию внутреннее представление понятия может также достраиваться.

Каждое предложение понятия строится из языковых шаблонов, которые могли быть введены в предыдущих предложениях текущего понятия, которые были введены в понятиях, уже используемых в текущем понятии, или находятся в библиотеке понятий доступной в данный момент по умолчанию.

Библиотека понятий доступная в текущий момент работы системы по умолчанию называется средой текущей работы системы. В системе ЭЗОП должны быть средства выбора и настройки библиотеки понятий – среды работы системы.

Для организации взаимодействия системы ЭЗОП с другими системами в системе ЭЗОП должна поддерживаться функция взаимных переводов с языка системы ЭЗОП на языки систем Ontolingua и Mathematica.

3. Требования к интерфейсу

Для пользователя системы формализованное понятие представляется его текстом. При формировании текущего понятия пользователь создает текст этого понятия, используя стандартные средства многооконного редактора и словари.

Таким образом, интерфейс системы ЭЗОП должен выглядеть как стандартный интерфейс многооконного редактора со всеми стандартными функциями, который дополнен возможностями работы со словарями системы, и в котором выделены окна работы с текущим понятием.

В первой версии системы многооконный интерфейс системы реализуется стандартными средствами Windows. В последующих версиях предполагается реализовать интерфейс средствами стандартных браузеров интернет систем.

4. Требования к языку системы

4.1. Общие требования.

Язык — это ключевой компонент в каждой области знаний. Он складывается в результате деятельности специалистов, по мере того, как складываются и развиваются понятия данной области знаний. Поэтому язык систем, основанных на знаниях и претендующих на использование или на наполнение знаниями, должен позволять включать новые языковые конструкции, которые появляются в данной области знаний. Это свойство языка систем называется открытостью языка.

Язык ЭЗОП — один из компонентов взаимодействия пользователя с системой ЭЗОП. Этот язык является открытым языком представления знаний и языком программирования задач. Процесс программирования на языке ЭЗОП — это диалог с системой программирования ЭЗОП, в котором специалист формулирует постановки задач проблемной области, используя уже введенные понятия как модули, или вводит новые понятия и их свойства.

С точки зрения специалиста по проблеме, пользующегося языком ЭЗОП, средства этого языка служат для выражения определений понятий проблемной области, для выражения доказательств некоторых свойств введенных понятий, для постановок задач и задания запросов к системе представления знаний. Однако выражения на языке ЭЗОП имеют внутреннюю интерпретацию, которая может быть скрыта от пользователя.

4.2. Принципы шаблонного языка

Средства представления знаний должны обеспечить возможность специалистам по конкретным проблемам представлять понятия своей предметной области для автоматизированной обработки, выражать условия задач и вопросы о предметной области, используя введенные понятия.

Выражения языка ЭЗОП строятся с помощью расширяемого набора шаблонов. Шаблон языка ЭЗОП — это последовательность слов и переменных, зафиксированная в словаре системы. С каждым шаблоном в системе связывается операция, которая может изменить текущее понятие, вызвать вычисление выражений, стоящих вместо переменных, или вызвать обработку запроса к текущему состоянию понятия. Из шаблонов строятся шаблонные выражения языка. Простое шаблонное выражение — это шаблон без переменных. Сложное шаблонное выражение — это шаблон, у которого наместо переменных подставлены уже построенные выражения.

4.3. Словари системы

1. Словарь последовательностей символов конца предложения
2. Словарь последовательностей символов – скобок
3. Словарь типов символов языка
4. Словарь шаблонов
5. Словарь понятий

5. Основные подсистемы

1. Многооконный интерфейс-редактор.
2. Подсистема работы с библиотекой понятий и словарями.
3. Подсистема формирования текущего понятия.
 - Модуль представления первого предложения текста текущего понятия в виде списка типизированных слов.
 - Модуль грамматического разбора предложения и составления термина ех-выражения.
 - Модуль интерпретации ех-выражения и его смысловой допустимости.
 - Модуль вычисления (выполнения команды) ех-выражения и пополнения внутреннего представления понятия.
4. Подсистема взаимодействия с системами Ontolingua и Mathematica.

6. Основные выразительные средства представления понятий

Основной тезис, который принимается в подходе представления непроцедурных знаний, состоит в том, что любое непроцедурное знание представляется именованным набором областей (типов) и имен отображений между ними. Этот набор имен называется сигнатурой понятия. Кроме того, в представлении понятия вводятся соотношения (аксиомы), с помощью которых выражается смысл понятия.

К выразительным средствам относятся также:

- возможность выделения некоторых элементов в областях;
- задание на именах областей отношения область – подобласть.

Выразительные средства расширяются введением понятий:

- декартова произведения областей;
- области подобластей данной области;
- области функций из одной области в другую и т.д.

Примеры представления понятий на языке системы ЭЗОП

Пусть, к примеру, наша предметная область — это понятия школьного учебника физики. Первый пример текста определения соответствует понятию механического движения.

Будем считать, что в приведенных далее примерах формальных определений понятий заглавные и соответствующие им строчные буквы не различаются, то есть, например, слова *двиЖение* и *движение* будут считаться совпадающими.

Механическое движение предполагает наличие пространства, времени, тел и отображения, задающего в каждый момент времени для каждого тела положение в пространстве. Поэтому понятие механического движения формально можно представить в следующем виде.

Определение: ДВИЖЕНИЕ.

ПРОСТРАНСТВО, ВРЕМЯ, ТЕЛА — это области.

Место — это отображение.

Место: ТЕЛА \times ВРЕМЯ \rightarrow ПРОСТРАНСТВО.

При определении этого понятия были использованы следующие шаблоны:

"Определение: ?Имя_понятия" — шаблон определения имени понятия (с его помощью в систему вводится имя понятия ДВИЖЕНИЕ).

"?Список_областей — это области" — этим шаблоном вводятся перечисленные в списке имена, как имена областей (ТЕЛА, ВРЕМЯ, ПРОСТРАНСТВО).

"?Отображение — это отображение" — этим шаблоном задается имя отображения.

"?Обл1 \times ?Обл2" — задает декартово произведение областей (ТЕЛА \times ВРЕМЯ).

С помощью шаблона "?Отображение: ?Обл1 \rightarrow ?Обл2" — вводится отображение из Обл1 в Обл2 (Место: ТЕЛА \times ВРЕМЯ \rightarrow ПРОСТРАНСТВО).

Так как физические величины можно умножать и делить друг на друга, то для определения понятий физических единиц и физических величин нам понадобится понятие коммутативной группы.

Определение: ?G — коммутативная группа.

G — область.

Произведение: G \times G \rightarrow G.

Обратный: G \rightarrow G.

e — элемент области G.

Если x, y — переменные типа G, то введем шаблоны "?x•?y", "?x/?y", "?x^(-1)" типа G.

"Соотношения:"

Если x, y, z — любые элементы области G, то

$x \bullet y$ = произведение (x, y),

$x^{(-1)}$ = обратный (x),

$x / y = x \bullet (y^{(-1)})$,

$x \bullet (y \bullet z) = (x \bullet y) \bullet z$, "ассоциативность операции произведения",

$e \cdot x = x, x \cdot e = x,$ " свойство единицы",
 $(x^{-1}) \cdot x = e, x \cdot (x^{-1}) = e,$ "свойства обратного элемента",
 $x \cdot y = y \cdot x,$ "коммутативность".

В этом примере с помощью уже известного шаблона "Определение: ?Имя_понятия" вводится имя понятия, которое в свою очередь является шаблоном с переменной ?G. В следующем предложении шаблоном " ?имя — область " G делается областью. Операции "Произведение" и "Обратный" на области G задаются с помощью использованного в предыдущем примере шаблона "?Отображение: ?Обл1 → ?Обл2 ". Единица "e" в области G задается шаблоном "?имя_элемента — элемент области ?область". Шаблонами "Если ?условие, то ?действия", "?список_переменных — переменные области ?имя_области" и "введем шаблоны ?список_текстов_шаблонов типа ?область" вводятся новые шаблоны. Заметим, что переменные введенные внутри этого предложения являются локальными и действуют только внутри предложения. Текст, заключенный в двойные кавычки, воспринимается системой как одно слово и используется в определении либо для задания текстов шаблонов, либо как тексты комментариев. В последнем предложении вводятся соотношения коммутативной группы, при этом используются шаблоны: "Если ?условие, то ?действия", "?список_переменных — любые элементы области ?имя_области", "скобки"(?"выражение") и "?выражение1=?выражение2 ". С помощью этих шаблонов вводятся уравнения, в которых переменные связаны кванторами всеобщности. Используя предыдущее определение, вводится новое понятие.

Определение: ?G— аддитивная группа.

G— коммутативная группа.

Переименование шаблонов:

Произведение на "Сумма",

Обратный на "Противоположный",

e на "0",

$?x \cdot ?y$ на " $?x + ?y$ ",

$?x^{-1}$ на " $-\?x$ ",

$?x/?y$ на " $?x-?y$ ".

В этом примере шаблоном "Переименование шаблонов: ?список_пар_шаблонов" внутри понятия заменяются одни шаблонов другими.

Теперь мы готовы с помощью описанных шаблонов и понятий дать определение физических единиц и физических величин.

Определение: Физические величины.

Используются: Числа.

Физические единицы — коммутативная группа.

Физические величины — коммутативная группа.

Если r_1 — переменная типа Числа, e_1 — переменная типа Физические единицы, x — переменная типа Физические величины, то введем шаблоны " $?r_1 ?e_1$ ", " $?r_1 \cdot ?x$ " типа Физические величины.

Если r_1, r_2 — любые элементы области Числа; e_1, e_2 — любые элементы области Физические единицы, то

$(r_1 e_1) \cdot (r_2 e_2) = (r_1 \cdot r_2) (e_1 \cdot e_2),$

$$(r_1 e_1) / (r_2 e_2) = (r_1 / r_2) (e_1 / e_2),$$

$$(r_1 e_1)^{-1} = r_1^{-1} e_1^{-1},$$

$$r_1 \cdot (r_2 e_1) = (r_1 \cdot r_2) e_1.$$

В этом примере операция шаблон "?r1 ?e1" строит по числу и физической единице физическую величину. Например, в выражении "2 м" двойка является элементом области Числа, а м — элемент области Физические единицы. Тогда "2 м" — это физическая величина.

Далее, так как физические величины одного типа можно складывать и по сложению они образуют аддитивную группу, то определение физической величины конкретного типа можно задать следующим определением.

Определение: ?X — физическая величина.

Используются: Физические величины.

X — аддитивная группа.

X — подобласть области Физические величины.

Введем шаблон "Единицы of ?X" типа область.

Единицы of X — подобласть области Физические единицы.

Если r1, r2 — любые элементы области Числа; e1 — любые элементы области Единицы of X, то

r1 e1 — элемент области X,

$$(r_1 e_1) + (r_2 e_1) = (r_1 + r_2) e_1.$$

Это определение построено с помощью шаблонов, которые были уже описаны в предыдущих примерах. В определении строится также новый шаблон "Единицы of ?X", который задает в области единиц физических величин подобласть для единиц физической величины X.

Используя понятия, введенные раньше, дадим определение движения тела.

Определение: ?тело движется.

Используется: ДВИЖЕНИЕ.

Тело — элемент области ТЕЛА.

ВРЕМЯ — физическая величина.

Час, мин, с — элементы области единицы of ВРЕМЯ.

$$1 \text{ час} = 60 \text{ мин. } 1 \text{ мин} = 60 \text{ сек.}$$

Если t1, t2 — переменные типа время, то введем шаблоны "промежуток времени от ?t1 до ?t2", "t (?t1, ?t2)" типа ВРЕМЯ.

Если t1, t2 — любые элементы области ВРЕМЯ, то промежуток времени от t1 до t2 = t (t1, t2),

$$t (t_1, t_2) = t_2 - t_1.$$

Если x — переменная типа тела, то введем шаблоны

"момент начала движения of x", "момент окончания движения of x", "время движения of x" типа ВРЕМЯ.

Момент начала движения of тело - момент окончания движения of тело = время движения of тело.

Длина — физическая величина.

м, км — элементы области единицы of длина.

$$1 \text{ км} = 1000 \text{ м.}$$

Если t_1, t_2 — переменные типа ВРЕМЯ, то введем шаблоны "длина пути of тело от t_1 до t_2 ", " S (тело, t_1, t_2)" типа длина.

Если t_1, t_2, t_3 — любые элементы области ВРЕМЯ, то

длина пути of таёй от t_1 до $t_2 = S$ (тело, t_1, t_2),

S (тело, t_1, t_2) = - S (тело, t_2, t_1),

S (тело, t_1, t_2) + S (тело, t_2, t_3) = S (тело, t_1, t_3).

В этом примере с помощью ранее описанных шаблонов вводятся физические величины времени, длины, физические единицы для них и основные соотношения между физическими единицами. При этом, например, предложение: "ВРЕМЯ — физическая величина", получено из шаблона имени предыдущего определения подстановкой вместо переменной X слова ВРЕМЯ. По этому предложению в текущее определение загружается предыдущее определение с заменой переменной X на ранее использованный термин ВРЕМЯ. В этом же определении вводятся понятия длины пути тела и времени движения тела.

Теперь дадим определение понятия равномерного движения с использованием ранее построенного понятия.

Определение: ?тело равномерно движется.

Тело движется.

Скорость — физическая величина.

Если s — любой элемент области единица of длина, t — любой элемент области единица of время, то s/t — единица of скорость.

Введем шаблоны: "скорость of тело", " V (тело)" типа скорость.

Скорость of тело = V (тело).

"Введем основное уравнение понятия".

Если t_1, t_2 — любые элементы области ВРЕМЯ, то

S (тело, t_1, t_2) = V (тело) * t (t_1, t_2).

Наконец можно использовать ранее определенные понятия для формулирования следующей задачи.

Определение: Задача 1.

Велосипедист равномерно движется.

t_1, t_2, t_3 — элементы области ВРЕМЯ.

Промежуток времени от t_1 до $t_2 = 10$ сек.

Длина пути of велосипедист от t_1 до $t_2 = 5$ м.

Длина пути of велосипедист от t_1 до $t_3 = 15$ м.

Чему равно $t(t_1, t_3)$?

В формулировке данной задачи первое предложение получено из шаблона "?тело равномерно движется" подстановкой на место переменной слова *велосипедист*. По этому предложению во внутреннее представление мира задачи переносится внутреннее представление предыдущего определения, где вместо термина *тело* подставлен новый термин *велосипедист*.

Второе предложение задачи строится из шаблона "?список_элементов — элементы области?имя_области", где на место списка элементов подставлен список t_1, t_2, t_3 имен конкретных моментов времени, вместо имени области — термин ВРЕМЯ. По этому

предложению в мире задачи в области ВРЕМЯ строятся три новых элемента с именами t_1 , t_2 , t_3 .

Три следующих предложения задачи — это уравнения, которые задают некоторые конкретные значения для некоторых выражений в мире задачи.

Последнее предложение — вопрос, построенный с помощью шаблона: "Чему равно ?Выражение ?". С этим шаблоном связывается действие выдачи на экран дескриптора (наиболее простого выражения), равного выражению, подставленному вместо переменной "?Выражение". То есть в ответ на этот вопрос должно быть выдано самое простое выражение, равное $t(t_1, 2)$ в мире задачи.

Можно сформулировать и другие задачи с другими вопросами. Можно таким же образом ввести понятия равноускоренного движения или движения навстречу друг другу и сформулировать задачи, где они будут использоваться.