

Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет Информационных технологий и управления
Кафедра Интеллектуальных информационных технологий

ОТЧЁТ
по ознакомительной практике

Выполнил:

Ф. И. Шаров

Студент группы
321701

Проверил:

В. Н. Тищенко

Минск 2024

СОДЕРЖАНИЕ

Введение	3
1 Постановка задачи	4
2 Формализация логико-семантической модели комплекса ostis- систем автоматизации проектирования решателей задач ostis-систем	5
3 Формальная семантическая спецификация библиографических ис- точников	8
Заключение	9
Список использованных источников	10

ВВЕДЕНИЕ

Цель:

Закрепить практические навыки формализации информации в интеллектуальных системах с использованием семантических сетей.

Задачи:

- Построение формализованных фрагментов теории интеллектуальных компьютерных систем и технологий их разработки;
- Построение формальной семантической спецификации библиографических источников, соответствующих указанным выше фрагментам;
- Оформление конкретных предложений по развитию текущей версии Стандарта интеллектуальных компьютерных систем и технологий их разработки

1 ПОСТАНОВКА ЗАДАЧИ

Часть 5 Учебной дисциплины "Представление и обработка информации в интеллектуальных системах"

⇒ библиографическая ссылка*:

- Грибова В.В..*КомплСППРиСРдСсОБ-2020ст*
:= [Комплекс средств поддержки процессов разработки и сопровождения решателей для систем с онтологическими базами знаний]
- Артемьева И.Л..*КонцеОдРРЗнОМО-2005ст*
:= [Концепция оболочки для разработки решателей задач на основе моделей онтологий]
- Заливако С.С..*СеманТКПИРЗ-2012ст*
:= [Семантическая технология компонентного проектирования интеллектуальных решателей задач]
- Голенков В.В..*ТехноКПЖЦССИКСНП-2023кн*
:= [Технология комплексной поддержки жизненного цикла семантически совместимых интеллектуальных компьютерных систем нового поколения]

2 ФОРМАЛИЗАЦИЯ ЛОГИКО-СЕМАНТИЧЕСКОЙ МОДЕЛИ КОМПЛЕКСА OSTIS-СИСТЕМ АВТОМАТИЗАЦИИ ПРОЕКТИРОВАНИЯ РЕШАТЕЛЕЙ ЗАДАЧ OSTIS-СИСТЕМ

Интеллектуальный решатель прикладных задач

:= [Intelligent application problem solver]

⇒ *определение**:

[Интеллектуальный решатель прикладных задач – это программная система, имеющая средства адаптации метода решения задачи к изменениям знаний предметной области при фиксированной онтологии[PDF] Концепция оболочки для разработки решателей задач на основе моделей онтологий]

⇒ *библиографическая ссылка**:

- *Артемьева И.Л..КонцепцияОдРРЗиОМО-2005ст*

Технологии проектирования интеллектуальных решателей

:= [Technologies for designing intelligent solvers]

⇒ *разбиение**:

= {

- *модель интеллектуального решателя задач*
- *библиотека ip-компонентов (intelligent property components – компонентов интеллектуальной собственности) решателя*
- *система автоматизации проектирования*
- *методика проектирования интеллектуальных решателей задач*
- *help-системы поддержки проектирования интеллектуальных решателей задач*
- *система управления коллективным проектированием интеллектуальных решателей задач*

}

⇒ *включение**:

логические подходы

= {

- *классическая дедуктивная логика*
- *методы индуктивного вывода*
- *абдуктивный вывод*
- *нечеткие логики*
- *правдоподобные рассуждения*
- *логика умолчаний*
- *темпоральная логика*

}

⇒ *библиографическая ссылка**:

- *Заливако С.С..СеманТКПИРЗ-2012ст*

Онтологический решатель задач

:= [Ontology problem solver]

⇒ *определение**:

[Онтологический решатель отвечает за процесс решения задач с использованием семантической модели предметной области, представленной в виде онтологии.]

⇒ *библиографическая ссылка**:

- *Грибова В.В..КомплСППРиСРдСсОБ-2020ст*

Проектирование онтологического решателя из программных единиц

:= [Designing an ontology solver from software units]
определение типов ПрЕд, описание их взаимодействия, реализация логики ПрЕд,
интеграция ПрЕд в единую систему и тестирование/отладка решателя

⇒ разбиение*:
программные единицы (ПрЕд)
= { • единицы разных типов (*Unitm*)
⇒ разбиение*:
доступу
= { • с доступом к информационным ресурсам (*ИнфРес*)
системы базовых знаний (*СБЗ*)
• без доступа к *ИнфРес*
}
• ПрЕд для вывода (промежуточного заключения)
• ПрЕд для поиска фактов
• ПрЕд для вычислений
• ПрЕд для связи с внешним окружением
}

⇒ примечание*:
[В составе решателя также могут быть и другие ПрЕд, не нуждающиеся в доступе к
ИнфРес – вычислительные, интерфейсные.]

⇒ библиографическая ссылка*:
• Грибова В.В..КомплСППРиСРдСсОБ-2020ст

Система автоматизации проектирования решателей задач *ostis*-систем

:= [Automation system for the design of problem solvers *ostis*-systems]
:= [ostis - система]
⇒ разбиение*:
модель системы
= { • *sc*-модель базы знаний
• *sc*-модель объединенного решателя задач
• *sc*-модель пользовательского интерфейса
}
⇒ разбиение*:
подсистемы
= { • подсистема автоматизации проектирования агентов обработки
знаний (*sc*-агентов)
• подсистема автоматизации проектирования *scr*-программ
}
⇒ разбиение*:
использование
= { • как подсистема в рамках *Метасистемы OSTIS*
⇒ пояснение*:
[Данный вариант использования предполагает отладку необходимых
компонентов в рамках *Метасистемы OSTIS* с последующим переносом
их в дочернюю *ostis*-систему]
• Как самостоятельная *ostis*-система
⇒ пояснение*:
[Предназначенная исключительно для разработки и отладки компо-
нентов решателей задач. В этом случае проектируемые компоненты
отлаживаются в рамках такой системы, а затем должны быть пере-

- несены в дочернюю ostis-систему]
 - *Как подсистема в рамках дочерней ostis-системы*
 - ⇒ *пояснение**:
[В таком варианте отладка компонентов осуществляется непосредственно в той же системе, в которой предполагается их использование, и дополнительного переноса не требуется.]
 - }
 - ⇒ *библиографическая ссылка**:
 - *Голенков В.В. ТехноКПЖЦССИКСНП-2023кн*

Отношение scr-программа поиска некорректности в scr-программе*

:= [Relationship scr-program to find incorrectness in scr-program*]

⇒ *пояснение**:

[Связывает класс некорректностей в scrпрограмме и scr-программу, которая может использоваться для выявления соответствующей некорректности в какой-либо другой scr-программе.

Указанная scr-программа должна иметь единственный параметр, который является in-параметром]

⇒ *разбиение**:

в зависимости от соответствующего класса некорректностей в scr-программе

= { • *scr-программу*

⇒ *пояснение**:

[В случае выявления некорректности в scr-программе вообще или ошибки в scrпрограмме на уровне программы.]

• *scr-процесс*

⇒ *пояснение**:

[Являющийся ключевым sc-элементом данной scr-программы в случае выявления ошибки в scrпрограмме на уровне множества параметров.]

• *множество операторов данной scr-программы*

⇒ *пояснение**:

[В случае выявления ошибки в scr-программе на уровне множества операторов]

• *знак конкретного scr-оператора*

⇒ *пояснение**:

[В случае выявления ошибки в scr-программе на уровне оператора.]

• *sc-дугу принадлежности*

⇒ *пояснение**:

[В случае выявления ошибки в scr-программе на уровне операнда.]

}

⇒ *примечание**:

[Если в результате верификации scr-программы выявлена некорректность, то формируется соответствующая структура и генерируется связка отношения некорректность в scr-программе*.]

⇒ *библиографическая ссылка**:

- *Голенков В.В. ТехноКПЖЦССИКСНП-2023кн*

3 ФОРМАЛЬНАЯ СЕМАНТИЧЕСКАЯ СПЕЦИФИКАЦИЯ БИБЛИОГРАФИЧЕСКИХ ИСТОЧНИКОВ

Грибова В.В..КомплСППРиСРдСсОБ-2020ст

⇒ аннотация*:

[Использование онтологического подхода является одним из современных подходов к созданию систем с базами знаний. Для построения жизнеспособных программных сервисов, работающих с такими базами знаний, и управления их коллективной разработкой предложена инструментальная среда. Метод инструментальной поддержки нацелен на создание и развитие библиотек онтолого-базируемых операций, на использование их при конструировании программных средств, на распределение полномочий по созданию компонентов систем с базами знаний, на контроль и интеграцию их в облачную сопровождаемую систему поддержки принятия решений на основе знаний.]

⇒ ключевой термины*:

- онтологическая база знаний
- алгоритм решения
- решатель задачи
- повторное использование
- сопровождение
- коллективная разработка

Артемьева И.Л..КонцеОдРРЗнОМО-2005ст

⇒ аннотация*:

[Интеллектуальный решатель прикладных задач – это программная система, имеющая средства адаптации метода решения задачи к изменениям знаний предметной области при фиксированной онтологии. В работе рассматривается подход к разработке интеллектуальных решателей задач, который состоит в формировании методов решения задач по моделям онтологии предметных областей и моделям онтологии задач для предметных областей, в которых онтология описывает не только структуру действительности, но и структуру знаний.]

Заливако С.С..СеманТКПИРЗ-2012ст

⇒ аннотация*:

[В работе приводится описание открытой семантической технологии проектирования интеллектуальных решателей задач. Отдельное внимание уделяется методике проектирования решателей и операций, являющихся составными частями таких решателей. Также в работе рассмотрено несколько примеров использования технологии при проектировании конкретных интеллектуальных систем по различным предметным областям.]

⇒ ключевой термины*:

- интеллектуальная система
- интеллектуальный решатель задач
- логический вывод

ЗАКЛЮЧЕНИЕ

Кратко о том, что сделано, в каком объеме, достигнуты ли цели и решены ли задачи

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [1] Артемьева, ИЛ. Концепция оболочки для разработки решателей задач на основе моделей онтологий / ИЛ Артемьева, ДА Крылов. — 2005. — Vol. 3. — P. 109–116.
- [2] Голенков, ВВ. Технология комплексной поддержки жизненного цикла семантически совместимых интеллектуальных компьютерных систем нового поколения / ВВ Голенков. — Беспринт, 2023.
- [3] Грибова, Валерия Викторовна. Комплекс средств поддержки процессов разработки и сопровождения решателей для систем с онтологическими базами знаний / Валерия Викторовна Грибова, Елена Арефьевна Шалфеева. № 4 (20). — Федеральное государственное бюджетное учреждение науки «Институт систем . . . , 2020. — P. 34–43.
- [4] Заливако, СС. Семантическая технология компонентного проектирования интеллектуальных решателей задач / СС Заливако, ДВ Шункевич. — БГУИР, 2012.