

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:

И. А. Бачило

Студент группы  
321703

Проверил:

В. Н. Тищенко

Минск 2024

## СОДЕРЖАНИЕ

Введение . . . . .	3
1 Постановка задачи . . . . .	4
2 Формализованные фрагменты теории Реализация sc-памяти в Программной платформе ostis-систем . . . . .	5
3 Формальная семантическая спецификация библиографических источников . . . . .	11
Заключение . . . . .	13
Список использованных источников . . . . .	14

## **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 2 Учебной дисциплины "Представление и обработка информации в интеллектуальных системах"**

⇒ библиографическая ссылка\*:

- Стандарт OSTIS
- Толковый словарь по Искусственному интеллекту

⇒ URL\*:

[<http://raai.org/library/tolk/aivoc.html>]

- ...

⇒ аттестационные вопросы\*:

- ⟨ • Вопрос 1 по Части 2 Учебной дисциплины "Представление и обработка информации в интеллектуальных системах"
- Вопрос 2 по Части 2 Учебной дисциплины "Представление и обработка информации в интеллектуальных системах"
- ...

## **Вопрос 3 по Части 6.3 Учебной дисциплины "Представление и обработка информации в интеллектуальных системах"**

:= [Реализация sc-памяти в Программной платформе ostis-систем ]

⇒ библиографическая ссылка\*:

- Зотов Н.В. Колич ПЭОнО-2023ст
- Шункевич Д.В. ВзаимАППО-2016ст

## **Вопрос 4 по Части 6.3 Учебной дисциплины "Представление и обработка информации в интеллектуальных системах"**

:= [Программный интерфейс Реализации sc-памяти в ostis-платформе ]

⇒ библиографическая ссылка\*:

- Зотов Н.В. Модел УПвОС-2023ст

## **Вопрос 5 по Части 6.3 Учебной дисциплины "Представление и обработка информации в интеллектуальных системах"**

:= [Реализация файловой памяти в Программной платформе ostis-систем ]

⇒ библиографическая ссылка\*:

- Зотов Н.В. Модел УПвОС-2023ст

## 2 ФОРМАЛИЗОВАННЫЕ ФРАГМЕНТЫ ТЕОРИИ РЕАЛИЗАЦИЯ SC-ПАМЯТИ В ПРОГРАММНОЙ ПЛАТФОРМЕ OSTIS-СИСТЕМ

### *Реализация sc-памяти ostis-платформы*

:= [Realization sc-memory ostis platform]

⇒ *пояснение\**:

[Текущая *Реализация sc-памяти ostis-платформы* позволяет хранить и представлять sc-конструкции, внешние информационные конструкции, не принадлежащие SC-коду, а также контролировать и согласовать процессы в ней.]

### *Программная платформа ostis-систем*

:= [Software platform ostis-systems]

:= [можно рассматривать как систему управления графовыми базами данных (СУБД)]

⇒ *примечание\**:

[ Однако она имеет ряд особенностей по сравнению с современными СУБД:  
• обрабатываемые графовые конструкции являются конструкциями SC-кода  
• в её основе заложены событийноориентированная обработка информации и основанный на ней многоагентный подход  
]

### *Программная реализация sc-памяти*

:= [Software implementation of sc-memory]

:= [понимается компонент программной платформы ostis-систем, осуществляющий хранение sc-конструкций и доступ к ним через соответствующий программный интерфейс [3].]

⇒ *решаемые задачи\**:

[  
• хранение sc-конструкций  
• хранение внешних по отношению к SC-коду информационных конструкций  
• доступ к sc-конструкциям, реализуемый через соответствующий программный интерфейс  
• управление процессами  
]

### *Программная реализация sc-памяти на ostis-платформе*

:= [Software implementation of the ostis platform sc-memory]

⇒ *пояснение\**:

[ В рамках данной *программной реализации ostis-платформы sc-память* представлена в виде набора фиксированного размера N, состоящего из сегментов ( $S = \langle s_1, s_2, \dots, s_i, \dots, s_n \rangle$ ), каждый из которых представляет собой набор фиксированного размера M, состоящей из элементов sc-памяти (или ячеек) ( $s_i = \langle e_{i1}, e_{i2}, \dots, e_{ij}, \dots, e_{im} \rangle, e_{ij} \in E$ ). Каждый элемент sc-памяти  $e_{ij}$  (в т.ч. элемент sc-памяти, соответствующий sc-узлу  $pk$ ) содержит:

- синтаксический класс
- семантический класс

уровни доступа соответствующего sc-элемента, выраженные в виде бинарных строк  $t$  и  $r$  соответственно, sc-адреса элементов sc-памяти, соответствующих первым входящему и выходящему sc-коннекторам, а также число входящих sc-коннекторов в заданный sc-элемент и число выходящих sc-коннекторов из заданного sc-элемента.

При этом каждый элемент sc-памяти, соответствующий sc-коннектору ch дополнительно содержит sc-адреса элементов sc-памяти, соответствующих начальному и конечному sc-элементам этого sc-коннектора, sc-адреса элементов sc-памяти, соответствующих предыдущим и следующим входящим и выходящим sc-коннекторам ( $E = N \ C, nk \ N, ch \ C$ ). ]

#### **Программная реализация файловой памяти**

- := [Software implementation of file memory]  
 := [обеспечивает хранение содержимого L файлов ostis-систем F ( $F \ N, N \times L \rightarrow F, F \times L \rightarrow F$ )]  
 ⇒ *пояснение\**:  
 [то есть *внешних информационных конструкций* (линейных текстов, изображений, видео и т.д.), а также выгрузку sc-памяти на дисковое пространство. В качестве структур данных в файловой памяти используются префиксные деревья и линейные списки.]

#### **Программный интерфейс реализации sc-памяти**

- := [Sc-memory implementation software interface]  
 := [представляет собой набор методов, позволяющих работать с sc-памятью.]  
 ⇒ *примечание\**:  
 [Его модель можно определить по формуле: ]  
 ⇒ *формула\**:  
 [

$$\begin{aligned}
 PI = & N^T \times C^{E \times E \times T} \times F^{N \times L} \times \\
 & \times \{E^{E \times \{EUT\} \times E} \cup E^{\{EUT\} \times T \times E} \cup E^{E \times T \times \{EUT\}}\} \times \\
 & \times T^E \times F^L \times L^F \times \{\top, \perp\}^E \quad (1)
 \end{aligned}$$

]

#### **Программная реализация подсистемы управления процессами в sc-памяти**

- := [Software implementation of the process control subsystem in sc-memory]  
 := [реализация sc-памяти]  
 ⇒ *разбиение\**:  
 { • *создание процессов*  
 • *чтение процессов*  
 • *удаление процессов*  
 • *синхронизацию процессов*  
 • *распределение sc-памяти между процессами*  
 }  
 ⇒ *примечание\**:  
 [В sc-памяти, обычно, используются потоки процессов. Поток может возникнуть в результате инициированного по событию sc-агента или в ответ на запрос к sc-памяти через сеть, выполняющих операции чтения и записи в ней.]

⇒ *пояснение\**:

[В предыдущих версиях программной платформы *ostis*-систем для обеспечения согласованного доступа к данным в разделяемой памяти (*sc*-памяти) применялись мьютексы и атомарные операции. В версии 0.9.0 для согласования доступа к структурам данным в *sc*-памяти были реализованы мониторы [3], а также методы для захвата и освобождения ресурсов для потоков-читателей  $P_r$  (далее – читателей) и потоков-писателей  $P_w$  (далее – писателей) ( $P = P_r \cup P_w$ ). Монитор  $m \in M$  состоит из: счётчика активных читателей, показывающего, активен ли в данный момент какой-либо писатель, очереди читателей и писателей, мьютекс, используемого для синхронизации доступа к этим элементам монитора. Очередь читателей и писателей представляет собой последовательность запросов на захват определенного ресурса. Каждый запрос включает в себя уникальный идентификатор потока, тип потока (читатель или писатель) и условную переменную, позволяющую обмениваться сообщениями между процессами (потоками). Эта очередь гарантирует, что ни один поток не останется «голодным». Важно отметить, что захват ресурса методом монитора для чтения одним потоком не блокирует захват этого же ресурса методом монитора для чтения другим потоком.]

⇒ *пояснение\**:

[Распределение *sc*-памяти писателям осуществляется посегментно с использованием специализированной хеш-таблицы  $Trps$ , которая позволяет определить, занят ли данный незаполненный сегмент *sc*-памяти другим писателем ( $s : S \rightarrow S_{ne}$ ,  $Trps : P_w \times S_{ne}$ ). Незаполненный сегмент *sc*-памяти может представлять собой как сегмент с невыделенными ячейками, так и сегмент с освобожденными ячейками. При распределении *sc*-памяти сначала происходит поиск незаполненных сегментов, которые не используются другими писателями. Если такие сегменты не найдены, то выделяются новые сегменты. Если в *sc*-памяти нет доступного места для новых сегментов, писатели могут использовать сегменты из списка занятых незаполненных сегментов. Для обеспечения согласованного доступа к сегментам для чтения каждый сегмент содержит уникальный монитор ( $sm : S \times M \rightarrow S_m$ ).]

⇒ *пояснение\**:

[кроме сегментов мониторы временно назначаются элементам *sc*-памяти и регистрируются событиями в ней. Мониторы элементов *sc*-памяти хранятся в специализированной хеш-таблице  $Tem$ . С помощью данных мониторов синхронизируется доступ к информации о сегменте, содержащейся в элементе *sc*-памяти ( $Tem : A \times M$ ). События  $v \in V$ , их типы  $tv \in Tv$ , *sc*-агенты, подписанные на них  $Av \in A$ , а также мониторы этих событий хранятся в единой хеш-таблице  $Tv$ . С помощью данных мониторов синхронизируются подписка и отписка на события через единую хеш-таблицу, а также инициирование самих *sc*-агентов ]

### *Действие в sc-памяти*

:= [Action in *sc*-memory]

⇒ *включение\**:

- *действие в sc-памяти, иницируемое вопросом*
- *действие редактирования базы знаний ostis-системы*
- *действие установки режима ostis-системы*
- *действие редактирования файла, хранимого в sc-памяти*
- *действие интерпретации программы, хранимой в sc-памяти*

### *Действие в sc-памяти, иницируемое вопросом*

:= [Action in *sc*-memory initiated by question]

- :=** [действие, направленное на формирование ответа на поставленный вопрос ]
- ⇒** включение\*:
- действие. сформировать заданный файл
  - действие. сформировать заданную структуру
    - ⇒** включение\*:
    - действие. верифицировать заданную структуру
      - ⇒** включение\*:
      - действие. установить истинность или ложность указываемого логического высказывания
      - действие. установить корректность или некорректность указываемой структуры
      - действие. сформировать структуру, описывающую некорректности, имеющиеся в указываемой структуре
  - действие. установить тип заданного sc-элемента
    - ⇒** включение\*:
    - действие. установить позитивность/негативность указываемой sc-дуги принадлежности или непринадлежности
    - действие. сформировать семантическую окрестность
      - ⇒** включение\*:
      - действие. сформировать полную семантическую окрестность указываемой сущности
      - действие. сформировать базовую семантическую окрестность указываемой сущности
      - действие. сформировать частную семантическую окрестность указываемой сущности
    - действие. сформировать структуру, описывающую связи между указываемыми сущностями
- ⇒** включение\*:
- действие. сформировать структуру, описывающую сходства указываемых сущностей
  - действие. сформировать структуру, описывающую различия указываемых сущностей
  - действие. сформировать структуру, описывающую способ решения указываемой задачи
  - действие. сформировать план генерации ответа на указанный вопрос
  - действие. сформировать протокол выполнения указываемого действия
  - действие. сформировать обоснование корректности указываемого решения
  - действие. верифицировать обоснование корректности указываемого решения
  - действие, одним из аргументов которого является некоторая обобщенная структура
  - действие, направленное на установление темпоральных характеристик указываемой сущности



- действие, направленное на установление пространственных характеристик указываемой сущности

### **Действие редактирования базы знаний**

**:=** [Knowledge base editing action]

**⇒** включение\*:

- действие. исправить ошибки в заданной структуре
- действие. преобразовать указанную структуру в соответствии с указанным правилом
- действие. отождествить два указанных sc-элемента
- действие. включить множество  
**:=** [ сделать все элементы множества  $s_i$  явно принадлежащими множеству  $s_j$ , то есть сгенерировать соответствующие sc-дуги принадлежности ]
- действие генерации sc-элементов  
**⇒** включение\*:  
  - действие генерации, одним из аргументов которого является некоторая обобщенная структура  
**⇒** включение\*:  
    - действие. сгенерировать структуру, изоморфную указываемому образцу
  - действие. сгенерировать sc-элемент указанного типа  
**⇒** включение\*:  
    - действие. сгенерировать sс-коннектор указанного типа
    - действие. сгенерировать sc-узел указанного типа
  - действие. сгенерировать структуру, содержащую указанные sc-элементы
  - действие. сгенерировать файл с заданным содержимым
  - действие. обновить понятия  
**:=** [ действие. заменить неиспользуемое понятие на его определение через используемое понятие ]
  - действие. установить указанный файл в качестве основного идентификатора указанного sc-элемента
  - действие. протранслировать содержимое указываемого файла в sс-память
  - действие. интегрировать указанную структуру в текущее состояние базы знаний
  - действие. сгенерировать структуру, описывающую историю эволюции ostis-системы
  - действие. сгенерировать структуру, описывающую историю эксплуатации ostis-системы
- действие удаления sc-элементов  
**⇒** включение\*:  
  - действие. удалить указанные sc-элементы  
**⇒** включение\*:  
    - действие. удалить указанный sc-элемент
    - действие. удалить sc-элементы, входящие в состав указанной структуры и не

- *являющиеся ключевыми узлами  
каких-либо сценариев  
действие. исключить указанные сценарии из  
клиентской части базы знаний*

### 3 ФОРМАЛЬНАЯ СЕМАНТИЧЕСКАЯ СПЕЦИФИКАЦИЯ БИБЛИОГРАФИЧЕСКИХ ИСТОЧНИКОВ

**Зотов Н.В. Модел УПвОС-2023ст**

⇒ *тип источника\**:

[научная статья]

⇒ *ключевой знак\**:

- Программная платформа *ostis-систем*
- Программная реализация *sc-памяти*
- Программная реализация *ostis-платформы sc-память*
- Программная реализация *файловой памяти*
- Программный интерфейс реализации *sc-памяти*
- Программная реализация подсистемы управления процессами в *sc-памяти*

⇒ *аннотация\**:

[В данной работе исследуется задача распределенной обработки информации в общей семантической памяти. Работа представляет новую модель управления процессами, отвечая на растущие требования по обработке информации в многопоточных средах и открытых вычислительных системах.]

⇒ *цитата\**:

[Программную платформу *ostis-систем* можно рассматривать как систему управления графовыми базами данных (СУБД) [1]. Однако она имеет ряд особенностей по сравнению с современными СУБД [1]]

**Зотов Н.В. Колич ПЭОнО-2023ст**

⇒ *тип источника\**:

[научная статья]

⇒ *ключевой знак\**:

- Реализация *sc-памяти ostisплатформы*

⇒ *аннотация\**:

[В данной работе проводится оценка эффективности общей семантической памяти программной платформы для проектирования и реализации интеллектуальных систем. Текущая работа описывает количественную оценку эффективности операций над общей семантической памятью в однопоточной и многопоточной средах.]

⇒ *цитата\**:

[Текущая Реализация *sc-памяти ostisплатформы* позволяет хранить и представлять *sc-конструкции*, внешние информационные конструкции, не принадлежащие *SC-коду*, а также контролировать и согласовать процессы в ней.]

**Шункевич Д.В. ВзаимАППО-2016ст**

⇒ *тип источника\**:

[научная статья]

⇒ *ключевой знак\**:

- Действие в *sc-памяти*
- Действие в *sc-памяти*, инициируемое вопросом
- Действие редактирования базы знаний

⇒ *аннотация\**:

[В работе рассматривается базовая модель построения машин обработки знаний систем, управляемых знаниями, построенных по открытой семантической технологии

⇒ проектирования интеллектуальных систем.]  
*цитата\**:  
[В рамках Технологии OSTIS sc-модель машины обработки знаний включает графодинамическую sc-память, в которой хранятся конструкции SCкода, и коллектив агентов взаимодействующих через эту память (sc-агентов).]

## **ЗАКЛЮЧЕНИЕ**

В ходе ознакомительной практики были приобретены ценные компетенции в сфере формализации текстового материала. Проведена тщательная работа по подбору соответствующей литературы, детальному разбору источников и выявлению ключевых элементов. Изучен теоретический базис Стандарта OSTIS с целью последующей интеграции собственной формализации. Кроме того, строго соблюдались синтаксические правила оформления формализованной теории.

В процессе практической деятельности были дополнены уже существующие формализованные концепты в монографии посредством добавления примечаний, разъяснений и конкретных примеров. Параллельно была проведена формализация дополнительной информации, касающейся формальной онтологии множеств, связей и отношений.

В результате выполнения ознакомительной практики были приобретены необходимые знания и умения в области формализации текстовых данных с соблюдением установленных стандартов и требований.

## **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

[1] Зотов, Н. В. Количественные показатели эффективности операций над общей семантической памятью интеллектуальных систем / Н. В. Зотов. — Кафедра ИИТ БГУИР, 2023. — С. 2.

[2] Зотов, Н. В. Модель управления процессами в общей семантической памяти интеллектуальных систем / Н. В. Зотов. — Кафедра ИИТ БГУИР, 2023. — С. 2.

[3] Шункевич, Д. В. Взаимодействие асинхронных параллельных процессов обработки знаний в общей семантической памяти / Д. В. Шункевич. — Кафедра ИИТ БГУИР, 2016. — С. 8.