

СЕМАНТИЧЕСКИЕ СРЕДСТВА РАЗРАБОТКИ ПРОГРАММНЫХ КОМПОНЕНТОВ НА БАЗЕ ИНТЕГРАЦИОННОЙ ПЛАТФОРМЫ

Ивашенко В. П.

Кафедра интеллектуальных информационных технологий, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: ivashenko@bsuir.by

В работе рассматривается применение интеграционной платформы для разработки программных компонентов на основе модели унифицированного семантического представления знаний.

ВВЕДЕНИЕ

Востребованность интеграции для решения задач в интеллектуальных системах обоснована в [1–2]. Основными видами интеграции [1] являются: вертикальная, горизонтально-фронтальная, горизонтально-профильная и непрерывная интеграция (Рис. 1).

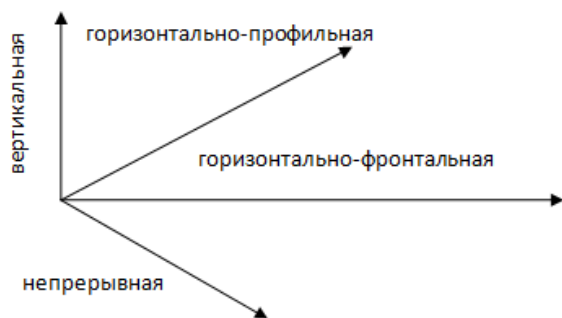


Рис. 1 – Направления интеграции

Решение задач вертикальной интеграции может быть обеспечено средствами модели обобщённых формальных языков [2], горизонтально-фронтальная интеграция обеспечивается коммуникацией и средствами унификации (например, языками модели унифицированного семантического представления знаний). Горизонтально-профильная и непрерывная интеграция обеспечиваются семантическими средствами используемых языков представления знаний и принципами обработки знаний [2]. Предлагаемые средства состоят из системы взаимодействующих программных компонент [1–3], использующих унифицированные модели семантического представления и обработки знаний [1–3] с целью интегрированной разработки новых программных компонент. Процесс их разработки сводится к их построению в результате обучения и генерации по образцу [4] путём семантического протоколирования [5] взаимодействий пользователя с системой [2].

ОБЩАЯ МОДЕЛЬ ПРЕДСТАВЛЕНИЯ И ОБРАБОТКИ ПРОГРАММНЫХ КОМПОНЕНТОВ

Общая модель представления и обработки программных компонент и библиотек компо-

нентов [3] строится на основе (реляционной модели) типологии (таксономии) компонент и команд операций их обработки.

Основными видами компонент являются:

1. Интерфейсно-диалоговые компоненты (ИДК).
2. Репрезентационные компоненты (РК).
3. Компоненты обработки потока знаний (ОПЗ).

Команды редактирования ИДК:

1. Создать ИДК.
2. Заменить сообщение ИДК.
3. Заменить конструирующую процедуру ИДК.
4. Заменить проверяющую процедуру ИДК.
5. Добавить ссылку на ИДК.
6. Удалить ссылку из ИДК.

Команды навигации для ИДК:

1. Получить сообщение ИДК.
2. Получить конструирующую процедуру ИДК.
3. Получить проверяющую процедуру ИДК.
4. Получить компоненты ИДК, на которые он ссылается.
5. Выбрать ИДК.

К РК относятся: протокол процесса взаимодействия с пользователем, процедуры, исходные тексты, библиотеки компонент (БК), онтологии: терминологические словари (образные представления), тезаурусы, таксономии, реляционные модели, (конъюнктивные) высказывания. При создании компонент-процедур протоколируются команды пользователя и результаты соответствующих операций над структурами знаний и данных. При этом используются соответствующие интерфейсно-диалоговые компоненты (Рис. 2).

Команды редактирования репрезентативных компонент (РК):

1. Начать создание РК указанного типа.
2. Завершить создание РК указанного типа.
3. Создать РК путём интеграции компонент.
4. Создать РК из части репрезентативного компонента.

5. Создать РК из фрагмента репрезентативного компонента протокола.
6. Преобразовать РК протокола в РК указанного типа.
7. Преобразовать РК в БК.
8. Добавить компонент в раздел БК.
9. Скопировать компоненты в новый подраздел из раздела БК.
10. Выгрузить компонент.
11. Загрузить компонент.

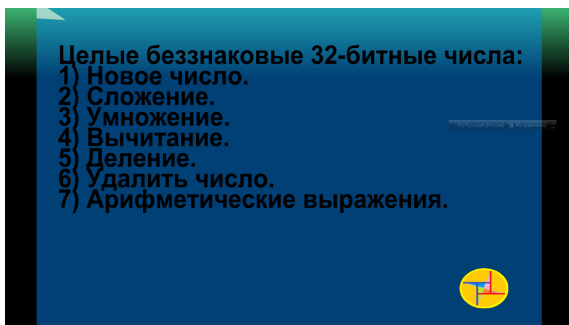


Рис. 2 – Арифметические команды над целыми 32-битными беззнаковыми числами

Команды навигации по БК:

1. Получить основной раздел БК.
2. Получить подразделы раздела БК.
3. Получить компоненты раздела БК.
4. Выбрать раздел БК.
5. Выбрать компонент БК.

Структура библиотеки исходных текстов может быть представлена внешним образом (json):

```
{"g" : {"n" : "2 8194"}, "test" : ["#./g/n", ...]}
```

или конструкцией в базе знаний [3] (Рис. 3).

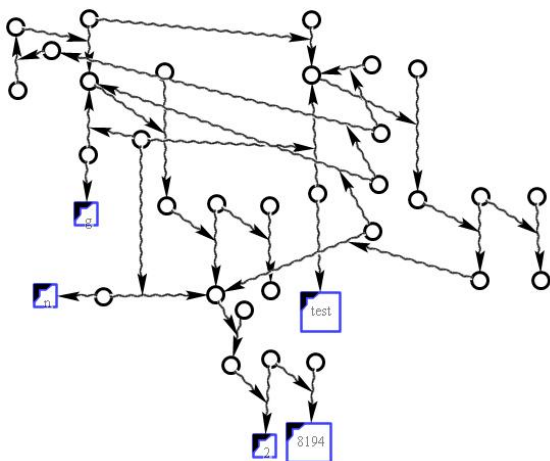


Рис. 3 – Структура библиотеки исходных текстов

Команды редактирования для компонентов ОПЗ:

1. Создать компонент ОПЗ.
2. Заменить процедуру компонента ОПЗ.
3. Установить компонент сообщения в качестве компонента ОПЗ.
4. Поменять потоки знаний разветвителя (компонента слияния).

5. Подключить компонент слияния к разветвителю.
6. Вставить компонент ОПЗ.
7. Изъять компонент ОПЗ.
8. Исключить систему компонентов второго потока знаний разветвителя (компонента слияния).

Команды навигации для компонентов ОПЗ:

1. Получить процедуру компонента ОПЗ.
2. Получить предшествующие компоненты компонента ОПЗ.
3. Получить последующие компоненты компонента ОПЗ.
4. Получить компонент сообщения.
5. Получить компонент ОПЗ.
6. Выбрать компонент ОПЗ.

Общими командами для компонент являются команды: получения типа компонента; получения авторов компонента; получения протокола преобразования компонента. Права на компонент определяются в соответствии с множеством автором и моделью управления доступом.

ЗАКЛЮЧЕНИЕ

Спроектирована общая модель представления и обработки программных компонентов и БК, использующих унифицированное семантическое представление знаний, способных обеспечить и ускорить разработку интеллектуальных систем на базе интеграционной платформы [1–2]. Развитие модели предполагает описание средств формирования развёрнутой спецификации компонентов, включая их детализированную типологию на основе семантики использующих компонент операций, требуемых для решения задач.

1. Ивашенко, В. П. Технология разработки программных компонентов интеллектуальных систем на основе интеграционной платформы / В. П. Ивашенко // Информационные технологии и системы 2021 (ИТС 2021) : материалы междунар. науч. конф. (Республика Беларусь, Минск, 24 ноября 2021 года) / редкол. : Л. Ю. Шилин [и др.]. – Минск: БГУИР, 2021. – С. 84–85.
2. Ivashenko, V. Application of an integration platform for ontological model-based problem solving using an unified semantic knowledge representation / V. Ivashenko // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS-2021) : сборник научных трудов / Белорусский государственный университет информатики и радиоэлектроники ; редкол. : В. В. Голенков [и др.]. – Минск, 2021. – Вып. 5. – С. 179–186.
3. Гаврилова, Т. А. Технология построения баз знаний на основе однородных семантических сетей / Т. А. Гаврилова, Н. А. Гулякина, В. П. Ивашенко // Информационные системы и технологии (IST2010). – Минск, 2010. – С. 450–453.
4. Waszkowski, R. Low-code platform for automating business processes in manufacturing // IFAC-PapersOnLine. – 2019. – Т. 52. – №. 10. – С. 376–381.
5. Ivashenko, V. Semantic Logging of Repeating Events in a Forward Branching Time Model / V. Ivashenko, N. Zotov, M. Orlov // Pattern Recognition and Information Processing (PRIP 2021). – Minsk, 2021. – P. 149–152.