

## НОВЫЕ ФУНКЦИОНАЛЬНЫЕ ВОЗМОЖНОСТИ СИСТЕМЫ ОБРАБОТКИ ЕСТЕСТВЕННОЯЗЫКОВЫХ СПЕЦИФИКАЦИЙ И СРЕДА ЕЕ ФУНКЦИОНИРОВАНИЯ

**Аннотация.** Описана среда функционирования языкового процессора, предназначенного для обработки моделей поведения систем, представленных на естественном языке. Данная среда обеспечивает настройку языкового процессора, основным этапом которой является построение синтаксической таблицы. Предложен подход к автоматизации построения продукций грамматики, предназначенных для пополнения синтаксической таблицы, используемой языковым процессором.

**Ключевые слова:** функциональная спецификация, языковой процессор, КС-грамматика, словарь ключевых слов.

### ВВЕДЕНИЕ

Настоящая статья является продолжением работы [1], в которой описана методика разработки функциональных спецификаций систем и устройств, а также программная система обработки естественных языковых спецификаций (ОЕС), предназначенная для анализа функциональных спецификаций (текстов требований) и преобразования их к формализованному виду, ориентированному на верификацию формальными методами [2]. Функциональные спецификации в инженерной практике проектирования систем и устройств — одно из средств описания модели поведения проектируемого объекта. Они определяют условия и правила переходов этого объекта из состояния в состояние и задаются как тексты на естественном языке. Важным этапом разработки систем и устройств является обоснование того, что спроектированный объект действительно выполняет функции, определенные в задании на разработку, и делает это правильно. Для этого строится формальная модель поведения проектируемого объекта, к которой применяются формальные методы верификации и валидации. Система ОЕС, предложенная в [1], предназначена для преобразования функциональных спецификаций на естественном языке в функциональные спецификации на формализованном языке. Предложения выходного языка имеют простой синтаксис, близкий к синтаксису базовых протоколов [2]:

$$\begin{aligned} P & ::= C \text{ ---} \rightarrow A. \\ C & ::= \text{Cond} \mid \text{Cond} \ \& \ C \\ A & ::= EA \mid EA; A \end{aligned}$$

Здесь  $P$  — предложение,  $A$  описывает реакцию проектируемого объекта в состоянии, удовлетворяющем условию  $C$ ,  $\text{Cond}$  — элементарная формула логики первого порядка,  $EA$  — присваивание или фраза естественного языка.

Дополнительно к описанной в [1] системе ОЕС создана единая среда (Оболочка ОЕС, или Оболочка), которая объединяет стандартные возможности работы с текстовыми файлами и специальные возможности настройки и использования языковых процессоров.

В настоящей работе описана Оболочка ОЕС, а также новые функции, разработка которых планировалась для развития системы ОЕС (см. [1]): средства автоматизации построения вспомогательной грамматики для перевода на формальный язык специальных фраз, называемых клише, из текста требований и средства работы со словарями ключевых слов.

## 1. ОБОЛОЧКА СИСТЕМЫ ОЕС

Оболочка является средой функционирования языкового процессора системы ОЕС и предназначена для его настройки, а также для работы с текстом требований, включая создание этого текста на естественном языке, предварительную его обработку и преобразование с помощью языкового процессора. Языковым процессором называем программу преобразования текста на одном языке (входном) в текст на том же или другом языке (выходном).

Оболочка ОЕС предоставляет пользователю такие возможности:

- стандартные средства работы с файлами (создать, открыть, сохранить, печатать);
- многооконный режим работы с подвижными границами некоторых окон и выделением активного окна;
- стандартные средства редактирования файлов (найти, выделить, копировать, вставить, удалить фрагмент текста в активном окне);
- структурирование исходного текста требований (расстановка структурных разделителей, обозначающих начало и конец каждого требования);
- средства настройки языкового процессора системы ОЕС;
- запуск языкового процессора системы ОЕС;
- средства работы со словарями ключевых слов.

Управление работой Оболочки осуществляется с помощью меню. Отдельные пункты меню и кнопки панели инструментов Оболочки обеспечивают обращение к языковому процессору и другим программам, объединенным в Оболочке. Для того чтобы обеспечить визуализацию и сохранение результатов работы языкового процессора, главное окно Оболочки разделено на несколько окон: левое, среднее, правое и нижнее (рис. 1). В левом окне находится входной текст языкового процессора; в среднее окно выводится выходной текст (результат работы языкового процессора); правое окно предназначается для отображения процесса работы языкового процессора (объем информации в этом окне определяется значениями параметров, которые задаются при запуске языкового процессора); в нижнем окне помещаются диагностические сообщения. Левое, среднее и правое окна могут

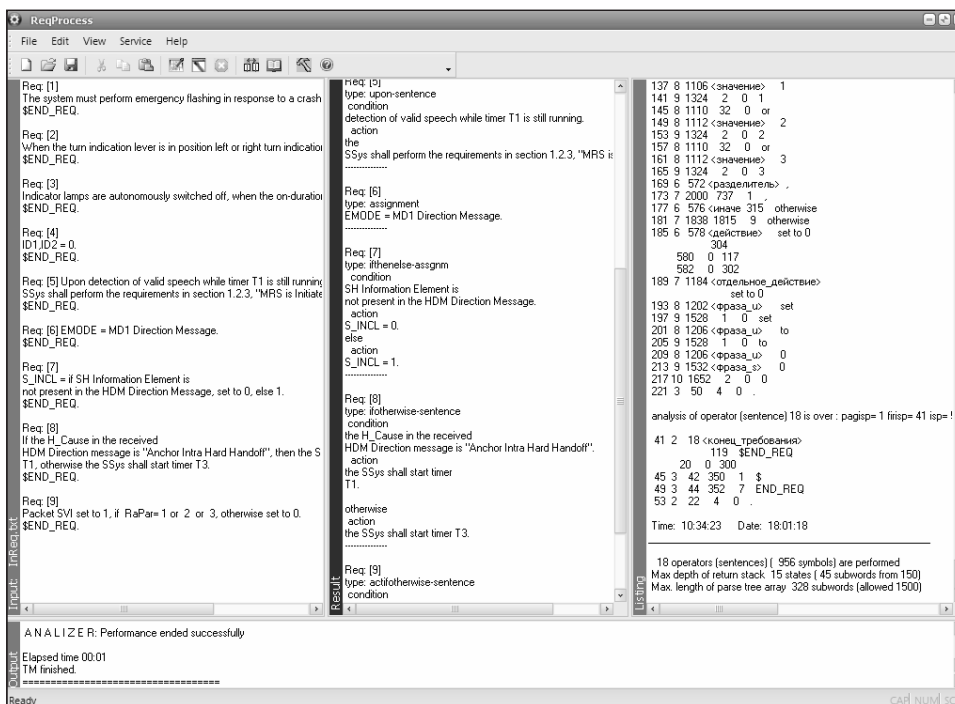


Рис. 1. Окна Оболочки. Пример входных и выходных данных Анализатора

быть активными. Окно становится активным, если установить курсор в его рабочей области (при этом окно может быть заполненным или пустым). Левая граница активного окна выделена цветом. Границы активного окна можно изменять с помощью курсора, а его содержимое — редактировать и сохранять под именем, указанным пользователем.

Используя активное окно, пользователь, не прерывая сеанса работы в Оболочке, имеет возможность редактировать входной текст, а затем запускать программы Оболочки с новым входным текстом.

Для задания параметров программ системы ОЕС, объединенных в Оболочке, предназначены специальные панели. На них имеются окна для ввода значений параметров и поля для задания режима формирования выходных данных. Такая организация панелей помогает пользователю осуществлять ввод значений параметров в полном объеме (не пропустив ни одного), а также избавляет его от необходимости указывать имена параметров. Специальная панель имеется и для работы со словарем.

## 2. ЯЗЫКОВОЙ ПРОЦЕССОР И ОСНОВНЫЕ ЭТАПЫ ЕГО РАЗРАБОТКИ

**2.1. Инструментальные средства.** Языковой процессор, функционирующий в Оболочке, разработан с помощью Расширяющейся Системы Программирования ТЕРЕМ (РСП ТЕРЕМ) [3], которая является инструментом для разработки гибких языковых процессоров.

РСП ТЕРЕМ содержит:

- язык для записи спецификаций языковых процессоров (LD);
- лексический анализатор (Сканер);
- модуль построения синтаксических таблиц (Конструктор);
- синтаксический анализатор (Анализатор);
- открытый набор универсальных семантических модулей («строительных блоков»), которые тиражируются в языковых процессорах различного назначения, создаваемых на основе РСП ТЕРЕМ;
- ряд вспомогательных модулей.

Назначение и структура составляющих РСП ТЕРЕМ описаны в [3].

**2.2. Этапы разработки языкового процессора.** Основой построения языкового процессора в РСП ТЕРЕМ является его спецификация (т.е. описание процессора, выполненное в соответствии с требованиями РСП ТЕРЕМ). Все этапы генерации языкового процессора связаны с подготовкой, обработкой и использованием его спецификации.

Разработка языкового процессора с помощью РСП ТЕРЕМ включает следующие шаги.

1. Разработка алгоритма перевода фраз с входного языка на выходной. Алгоритм строится как система так называемых семантических действий, т.е. правил перевода фраз входного языка, и задается в формализованном виде, выбранном разработчиком. Семантические действия именуются.

2. Описание синтаксиса входного языка с помощью нелеворекурсивной КС-грамматики.

3. Преобразование построенной грамматики входного языка в текст на языке LD (языке модифицированных Бэкусо-Науровских форм). Грамматика должна содержать нетерминальные символы, из которых выводятся такие фразы, перевод которых на выходной язык задан алгоритмом, разработанным на первом шаге. Кроме правил, описывающих синтаксис входного языка, в тексте на языке LD имеются правила для системных нетерминалов-параметров, где указываются: знак, являющийся разделителем предложений входного языка, аксиома грамматики и имя разрабатываемого языкового процессора.

4. Приписывание нетерминалам в LD-тексте имен соответствующих семантических действий.



Рис. 2. Конструктор таблиц

числа, разделители слов и др. Если необходимо выделить во входном языке класс лексем, не предусмотренный в системном сканере, то функция распознавания этого класса формируется из прототипов, имеющихся в системном сканере. Функции распознавания лексем, которых нет во входном языке, исключаются из системного сканера. Список всех классов лексем входного языка включается в LD-текст. Этот шаг завершает построение спецификации языкового процессора.

6. Разработка набора семантических действий. На этом этапе программируются процедуры выполнения семантических действий и встраиваются в специальный стандартный модуль РСП ТЕРЕМ.

7. Отладка спецификации языкового процессора. Выполняется с помощью модуля Конструктор. На этом этапе проверяется синтаксическая правильность спецификации языкового процессора и полнота описания нетерминальных символов, использованных в описании входного языка.

8. Построение синтаксической таблицы (СТ). Стандартный модуль Конструктор РСП ТЕРЕМ строит СТ по спецификации языкового процессора, не содержащей ошибок (рис. 2).

9. Отладка языкового процессора.

Перечисленные шаги работы выполняются циклически в процессе настройки и наращивания возможностей реализуемого процессора. Предусмотрены режимы вывода информации для отладки различной степени подробности.

Модуль Конструктор имеет два режима работы: построение новой СТ и пополнение существующей. Для пополнения ранее созданной СТ составляется спецификация на языке LD (так называемая вспомогательная грамматика).

### 3. ОСОБЕННОСТИ РАЗРАБОТКИ ЯЗЫКОВОГО ПРОЦЕССОРА СИСТЕМЫ ОЕС

**3.1. Многоуровневая обработка исходного текста и использование промежуточных языков.** Обработка текста в системе ОЕС состоит из нескольких этапов [1]. Первый этап — определение типов и структуры предложений исходного текста. Результат формируется на промежуточном языке, который содержит названия типов предложений, а также заголовки составных частей предложений. На втором этапе обрабатываются составные части предложений, выделенные ранее как «условия» и «действия». Поскольку первый этап выполняется за два шага, существует еще один промежуточный язык. Языковой процессор настраивается на обработку текстов как на входном, так и на промежуточных языках.

Проиллюстрируем работу языкового процессора с промежуточными языками на простом примере общего вида. Пусть в исходном тексте требований встречается предложение вида

Если Условие, то Действие1 или Действие2.

Здесь Условие, Действие1, Действие2 обозначают фразы на естественном языке. Результат первого шага первого этапа обработки такого предложения имеет на промежуточном языке такой вид:

```
type: ifor-sentence
```

```
    Если Условие, то Действие1.
    Если Условие, то Действие2.
```

После выполнения второго шага первого этапа обработки имеем результат следующего вида на промежуточном языке:

5. Модификация Сканера. Системный лексический анализатор РСП ТЕРЕМ (Сканер) содержит набор функций, распознающих такие классы лексем, как идентификаторы, константы, целые положительные

```

type: ifcompound-sentence
  condition
Условие.
  action
Действие1.
type: ifcompound-sentence
  condition
Условие.
  action
Действие2.

```

На втором этапе обработки имеем результат вида:

```

P_Y ----> P_D1.
P_Y ----> P_D2.

```

Здесь  $P_Y$ ,  $P_{D1}$ ,  $P_{D2}$  — результаты преобразования фраз *Условие*, *Действие1*, *Действие2* к формальному виду, как это показано в [1].

Организация многоуровневой обработки исходного текста требований дает возможность разделить спецификацию языкового процессора на несколько частей (соответствующих уровням обработки исходного текста). Таким образом переходим к работе с менее громоздкими спецификациями (языковых процессоров) и получаем возможность анализировать промежуточные результаты преобразования исходного текста.

**3.2. Задание словаря ключевых слов в качестве параметра.** РСП ТЕРЕМ обеспечивает построение языковых процессоров, использующих словарь ключевых слов. Однако при увеличении объема словаря работа языкового процессора замедляется. При разработке языкового процессора системы ОЕС обеспечена возможность задавать словарь ключевых слов в качестве параметра либо не использовать его. В результате достигается гибкость настройки языкового процессора на язык исходных текстов и на режим обработки (со словарем/без словаря).

**3.3. Разнообразие семантических действий.** Семантические действия, которые используются в языковых процессорах различного назначения, реализованы в базовых модулях РСП ТЕРЕМ. Их набор в процессе накопления опыта создания языковых процессоров с помощью этой системы пополняется.

Семантические действия выполняются языковым процессором при обходе дерева синтаксического анализа.

В языковом процессоре системы ОЕС используются семантические действия таких видов:

- семантические шаблоны вида  $@"01\text{текст}"$ , где *текст* — произвольная последовательность символов, которая вставляется в выходной файл;
- сокращенные шаблоны  $@"01@0"$ , где  $@0$  обозначает использование в качестве перевода в вершине дерева синтаксического анализа той фразы, которая выводится из данной вершины;
- вывод нетерминального символа в выходной файл;
- вывод в выходной файл специальных знаков для структурирования результирующего текста;
- согласованные действия для перевода предложений, содержащих такие фразы, что при переводе одной из них необходимо учитывать перевод другой, а для этого необходимо запоминать данные для последующего применения;
- запись фразы в результат при перестановке, когда требуется переставить части входной фразы и вывести в результат перевод первой части фразы после перевода ее второй части;
- перевод предложения, в котором используется перечислимый тип данных;
- перевод составного присваивания;
- запись результата перевода в специальный вспомогательный файл при порождении вспомогательной грамматики;
- вывод содержимого вспомогательного файла в результат при порождении вспомогательной грамматики.

Итак, информационная среда семантических программ состоит из данных, которые предоставляются синтаксической подсистемой через дерево анализа, параметров семантических действий и результатов уже выполненных семантических программ.

Программы выполнения семантических действий первых двух видов из приведенного списка содержатся в базовых модулях РСП ТЕРЕМ. Программы выполнения остальных видов действий созданы при разработке языкового процессора системы ОЕС.

Дополнительные действия потребовались для вывода нетерминального символа в выходной текст на промежуточном языке. Особый раздел образуют так называемые согласованные действия. Они применяются в совокупности, если при переводе предложения необходимо запомнить часть предложения или результат перевода части предложения, чтобы затем использовать запомненное при выполнении последующих семантических действий, применяемых при переводе этого предложения. Такие действия нужны, например, когда в выходной файл следует записать результат перевода первой части предложения после результата перевода его второй части.

При обработке предложения, в котором используется перечислимый тип данных, необходимо не только перевести предложение, но и обеспечить порождение нескольких предложений в зависимости от числа значений перечислимого типа. При этом перевод выполняется за один обход дерева анализа.

Аналогично для составного присваивания (когда в левой части присваивания указан список переменных) в результате перевода порождается соответствующее количество простых присваиваний.

**3.4. Использование вспомогательной грамматики.** Результат перевода исходного текста требований на формализованный язык с помощью системы ОЕС формируется на втором этапе обработки этого текста. При этом используется СТ, включающая слова и словосочетания из предметной области, к которой принадлежит исходный текст. Для того чтобы упростить построение этой СТ, используется способность РСП ТЕРЕМ пополнять готовые СТ дополнительными правилами и создается СТ из двух спецификаций, первая из которых (так называемая основная грамматика) неполная, а вторая (так называемая вспомогательная грамматика) ее дополняет. В основной грамматике сосредоточены правила общего вида для преобразования фраз исходного текста, а во вспомогательной — правила для обработки фраз-клише и сообщений, характерных для языка разрабатываемого проекта.

Результирующий языковой процессор настроен на обработку текста как на входном, так и на промежуточном языке, и на работу с различными словарями (рис. 3). На вход языкового процессора подается текст на одном из допустимых языков (естественном или промежуточном) либо текст описания атрибутов для построения вспомогательной грамматики, а также СТ. Пользователь имеет возможность выбирать режим работы процессора: со словарем (тогда на вход процессора подается словарь) или без него. Результат преобразования входного текста языковым процессором записывается в файл со специальным расширением и выводится на экран. Кроме выходного текста, выдается также листинг, содер-

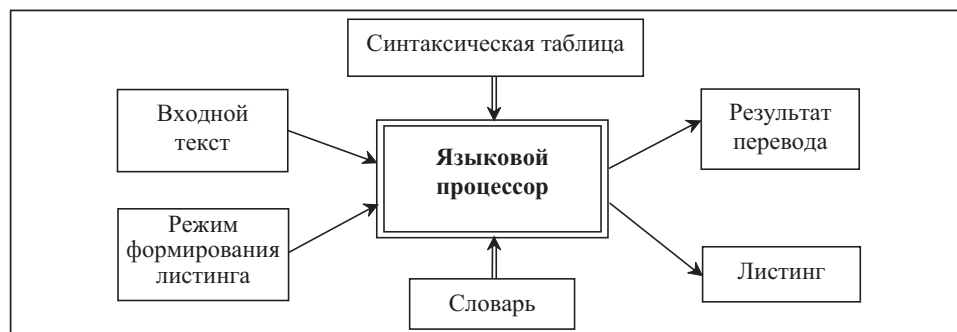


Рис. 3. Результирующий процессор

жащий информацию о работе языкового процессора; объем этой информации регулируется заданием режима. Листинг может содержать сведения о ходе грамматического разбора предложений входного текста (степень детализации которых определяется заданным режимом), информацию об ошибках во входном тексте, сведения об аварийных ситуациях при работе языкового процессора.

#### 4. СРЕДСТВА АВТОМАТИЗАЦИИ ПОСТРОЕНИЯ ВСПОМОГАТЕЛЬНОЙ ГРАММАТИКИ ДЛЯ ПРЕОБРАЗОВАНИЯ СПЕЦИАЛЬНЫХ ФРАЗ ИЗ ТЕКСТА ТРЕБОВАНИЙ К ФОРМАЛЬНОМУ ВИДУ

Методика разработки функциональных спецификаций систем и устройств, предложенная в [1], предполагает построение информационной модели проектируемого объекта. На этапе разработки этой модели определяются и описываются атрибуты объекта. На практике довольно часто используются атрибуты булевого, перечислимого типов и так называемые «сообщения». Предлагаем следующую специальную форму описания таких атрибутов:

```
<описания атрибутов> ::= <раздел> { <раздел> } ;
<раздел> ::= <атрибуты булевого типа>
| <атрибуты типа сообщение>
| <атрибуты перечислимого типа> ;
<атрибуты булевого типа> ::= <заголовок булевого типа>
<описание атрибута булевого типа> { <разделитель> <описание
атрибута булевого типа> } . ;
<заголовок булевого типа> ::= Атрибуты булевого типа : ;
<описание атрибута булевого типа> ::= <идентификатор> = 1
означает, что <описание>
| <идентификатор> = 0 означает, что <описание> ;
<описание> ::= <слово> { <слово> } ;
<атрибуты типа сообщение> ::= <заголовок сообщений>
<описание атрибута типа сообщение>
{ <разделитель> <описание атрибута типа сообщение> } . ;
<заголовок сообщений> ::= Перечень сообщений : ;
<описание атрибута типа сообщение> ::= <идентификатор> -
<описание> ;
<атрибуты перечислимого типа> ::= <заголовок перечислимого типа>
<описание атрибута перечислимого типа>
{ <разделитель1> <описание атрибута перечислимого типа> } . ;
<заголовок перечислимого типа> ::= Атрибуты перечислимого типа : ;
<описание атрибута перечислимого типа> ::= <заголовок> <строка
описания>
{ <разделитель1> <строка описания> } ;
<заголовок> ::= <имя типа> = ( <идентификатор> { ,
<идентификатор> } ) <разделитель1> ;
<строка описания> ::= <идентификатор> - <описание атрибута> ;
<описание атрибута> ::= <фраза> { <разделитель> <фраза> } ;
<фраза> ::= <слово> { <слово> } ;
<имя типа> ::= <идентификатор> ;
<разделитель> ::= , ;
<разделитель1> ::= ; ;
```

Нетерминальный символ слово здесь не описан. Его определение известно языковому процессору системы ОЕС.

Разработаны и реализованы семантические действия для преобразования описания атрибутов, построенного в соответствии с предложенной формой, во вспомогательную грамматику. Языковой процессор системы ОЕС настроен на преобразование описания атрибутов, построенного по данному образцу, во вспомогательную грамматику. Она используется для пополнения СТ на втором этапе обработки текста требований средствами системы ОЕС. Фразы, обозначенные в данном описании как <описание> и <описание атрибута>, называем клише. В соответствии с методикой, описанной в [1], эти клише используются при построении функциональных спецификаций (требований к поведению) проектируемого объекта. Языковой процессор, который применяет порожденную по описанию атрибутов вспомогательную грамматику, заменяет клише формальными выражениями.

Проиллюстрируем в общем виде процесс образования клише и его преобразования с помощью вспомогательной грамматики. Рассмотрим составленный по предложенной форме простой текст описания атрибутов следующего вида:

Атрибуты перечислимого типа:

ИмяТипа = (Значение1, Значение2);

Значение1 - фраза1;

Значение2 - фраза2.

Здесь фраза1, фраза2 — обозначения фраз на естественном языке (так называемых фраз-клише), которые поясняют смысл соответствующих значений перечислимого типа ИмяТипа. По заданному тексту описания атрибутов автоматически строится вспомогательная грамматика (спецификация на языке LD), в которой для каждой фразы-клише имеются продукции, снабженные семантическими действиями, определяющими правила преобразования этой фразы-клише в зависимости от того, в какой части предложения-требования она встречается.

Пусть, например, во входном тексте требований встречается предложение вида «Если фраза1, Событие1, то фраза2, Событие2.»,

где фраза1, фраза2 — фразы-клише из описания атрибута перечислимого типа ИмяТипа, а Событие1, Событие2 — описания каких-либо событий. Тогда в результате работы языкового процессора системы ОЕС, использующего, кроме основной, вспомогательную грамматику, порожденную по заданному тексту описания атрибутов, получим выражение вида

(ИмяТипа = Значение1) & РПС1 ---> (ИмяТипа := Значение2); РПС2.

Здесь РПС1, РПС2 — результаты преобразования описаний событий Событие1 и Событие2 соответственно.

Чтобы обеспечить автоматическое порождение вспомогательной грамматики, использующейся для обработки фраз-клише, потребовалось пополнить набор семантических действий языкового процессора системы ОЕС.

## 5. СРЕДСТВА РАБОТЫ СО СЛОВАРИМИ КЛЮЧЕВЫХ СЛОВ

Для настройки работы системы ОЕС с текстом на том или ином естественном языке необходимо выбрать подходящий словарь ключевых слов. В словаре хранятся термины, слова и словосочетания из области, к которой принадлежит входной текст, а также слова, предназначенные для выделения грамматических конструкций. В экспериментах с системой ОЕС использовались тексты на украинском, русском и английском языках.

Словарь состоит из разделов, каждый из которых имеет голову и тело. Голова — это целое положительное число (уникальный номер раздела). Номер раздела обозначает тип слов, содержащихся в теле раздела. Слова (словосочетания) в словаре делятся на типы в зависимости от их роли в распознавании типов предложений и фраз входного текста. Раздел представляет собой последовательность строк.



Номер раздела записан в его первой строке. Каждая строка тела раздела содержит слово или словосочетание.

Оболочка предоставляет такие возможности работы со словарем:

- добавить раздел в словарь;
- удалить раздел из словаря;
- переименовать раздел;
- добавить строку (слово/словосочетание) в раздел;
- удалить строку (слово/словосочетание) из раздела;
- изменить содержимое строки в теле раздела;
- сохранить словарь под новым именем.

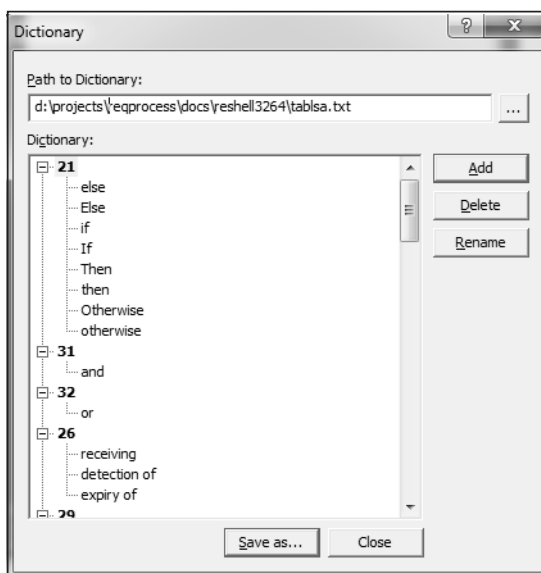


Рис. 4. Панель диалога для работы со словарями со словарями в Оболочке показана на рис. 4.

#### ЗАКЛЮЧЕНИЕ

Разработанная Оболочка является средой функционирования:

- языкового процессора системы ОЕС;
- программы создания синтаксических таблиц (Конструктора);
- программы работы со словарями ключевых слов;
- программы структурирования текста требований, написанного на естественном языке.

Преимущества использования Оболочки, которые проявились при ее опытной эксплуатации:

- экономия времени пользователя системы ОЕС при задании параметров и режимов работы программ;
- дополнительное средство контроля правильности задания параметров программ;
- обеспечение визуализации результатов работы программ Оболочки;
- возможность в течение одного сеанса (работы Оболочки) редактировать обрабатываемый текст, используя активное окно.

Предложенная модель раздела описаний атрибутов и разработанные семантические действия дали возможность автоматизировать порождение вспомогательной грамматики, предназначенной для обработки клише.

Использование клише, с одной стороны, позволяет формулировать функциональные спецификации на естественном языке, а с другой — облегчает их формализацию.

Средства автоматизации построения вспомогательной грамматики дают возможность использовать раздел описания атрибутов проектируемой системы для того, чтобы избавить пользователя от части рутинной работы, связанной с настройкой языкового процессора на предметную область.

Подготовка к работе с текстом требований к программной системе (устройству) в рамках описанной Оболочки состоит в разработке информационной модели проектируемого объекта, задании описания атрибутов булевого и перечислимого типа и типа «сообщение» в форме, описанной в разд. 4, пополнении словаря ключевых слов терминами из соответствующей предметной области, построении модели поведения проектируемого объекта в виде функциональных спецификаций на естественном языке. В дальнейшем можно воспользоваться имеющимся языковым процессором или при необходимости модифицировать его.

Одним из направлений развития Оболочки системы ОЕС является разработка и создание средств для ведения проекта с автоматическим переходом от одной стадии работы с текстом требований к следующей.

Планируется развивать систему ОЕС в следующих направлениях:

- разработка средств формализации новых видов предложений;
- пополнение промежуточных языков средствами описания структуры предложений, обрабатываемых языковым процессором;
- выделение тех частей спецификации языкового процессора, которые целесообразно вынести во вспомогательную грамматику.

#### СПИСОК ЛИТЕРАТУРЫ

1. Годлевский А.Б., Мищенко Н.М., Мороховец М.К., Фелижанко О.Д., Щёголева Н.Н. Методика разработки и средства формализации функциональных спецификаций систем и устройств. *Кибернетика и системный анализ*. 2016. Т. 52, № 5. С. 25–42.
2. Летичевский А.А., Капитонова Ю.В., Волков В.А., Летичевский А.А., Баранов С.Н., Котляров В.П., Вейгерт Т. Спецификация систем с помощью базовых протоколов. *Кибернетика и системный анализ*. 2005. № 4. С. 3–21.
3. Мищенко Н.М., Щёголева Н.М. О проектировании языковых процессоров на ПЭВМ. *Кибернетика и системный анализ*. 1993. № 2. С. 110–117.

Надійшла до редакції 20.01.2018

#### **Н.М. Міщенко, М.К. Мороховець, О.Д. Феліжанко, Є.В. Штелик, Н.М. Щоголева НОВІ ФУНКЦІОНАЛЬНІ МОЖЛИВОСТІ СИСТЕМИ ОБРОБКИ ПРИРОДНОМОВНИХ СПЕЦИФІКАЦІЙ ТА СЕРЕДОВИЩЕ ЇЇ ФУНКЦІОНУВАННЯ**

**Анотація.** Описано середовище функціонування мовного процесора, який призначено для обробки моделей поведінки систем, що подані натуральною мовою. Це середовище забезпечує налаштування мовного процесора, основним етапом якого є побудова синтаксичної таблиці. Запропоновано підхід до автоматизації побудови продукцій граматики, що призначені для поповнення синтаксичної таблиці, яку використовує мовний процесор.

**Ключові слова:** функціональна специфікація, мовний процесор, КВ-граматика, словник ключових слів.

#### **N.M. Mishchenko, M.K. Morokhovets, O.D. Felizhanko, Y.V. Shtelik, N.N. Shchogoleva**

#### **NEW FUNCTIONALITIES OF THE SYSTEM FOR NATURAL-LANGUAGE SPECIFICATIONS PROCESSING AND ITS OPERATING ENVIRONMENT**

**Abstract.** The paper describes the operating environment of the language processor intended for for handling behavior models of systems represented in natural language. This environment provides tuning of the language processor; the main stage of this tuning is construction of a syntax table. An approach is proposed for automating the construction of grammar rules intended for replenishment of the syntactic table that the language processor uses.

**Keywords:** functional specification, language processor, context-free grammar, keyword dictionary.

#### **Мищенко Надежда Михайловна,**

кандидат физ.-мат. наук, старший научный сотрудник, Киев, e-mail: nadmykh@ukr.net.

#### **Мороховец Марина Константиновна,**

кандидат физ.-мат. наук, старший научный сотрудник Института кибернетики им. В.М. Глушкова НАН Украины, Киев, e-mail: marina.morokhovets@gmail.com.

#### **Фелижанко Ольга Дмитриевна,**

научный сотрудник Института кибернетики им. В.М. Глушкова НАН Украины, Киев, e-mail: Felizh.olg@gmail.com.

#### **Штелик Евгений Васильевич,**

инженер-программист Института кибернетики им. В.М. Глушкова НАН Украины, Киев, e-mail: euhimer@gmail.com.

#### **Щёголева Наталья Николаевна,**

научный сотрудник Института кибернетики им. В.М. Глушкова НАН Украины, Киев, e-mail: natashch2904@gmail.com.