УДК 519.7

Г.Г. Четвериков, И.Д. Вечирская, А.С. Пузик

Харьковский национальный университет радиоэлектроники, Харьков

РАЗРАБОТКА ПРОГРАММНОЙ СИСТЕМЫ ЭЛЕКТРОННОГО ТРЁХЪЯЗЫЧНОГО СЛОВАРЯ

Статья посвящена разработке программной системы русско-украинско-английского терминологического словаря. Входными данными являются отсканированные и распознанные документы в формате MSWord. Проанализированы ошибки, возникающие при синтаксическом анализе входных данных, и указаны пути их устранения при помощи регулярных выражений. Приведена и детально описана схема лексикографической базы данных словаря, описаны классы модели данных и классы модели представления системы. Кроме этого, представлено подробное описание программной системы с точки зрения пользователя, а также обозначены перспективы использования как самого словаря, так и методов его построения. Программная система построена с использованием шаблона проектирования Model-View-ViewModel. Благодаря использованию данного шаблона интерфейс пользователя отделён от логики программы, что позволяет осуществлять независимые изменения отдельных частей программной системы. Разработанная программная система позволяет редактировать, наполнять и таким образом создавать новые тематические переводные электронные словари. Преимуществом системы является равноправие языков, т.е. в каждом отдельном случае пользователь назначает сам какой из языков будет главным.

Ключевые слова: *алгебра конечных предикатов, база данных, лексикография, лексическая единица, шаблон MVVM, парсинг, программная система.*

Введение

Задача построения электронного словаря только на первый взгляд может показаться несложной. На самом деле она представляет собой трудоёмкую многоэтапную процедуру, требующую в процессе решения применения порой даже нестандартных методов. Возникают проблемы, связанные с представлением, обработкой и хранением данных. Модель данных, которая используется в системе управления базы данных, должна учитывать аспект структуры, т.е. описание типов и структур данных в базе данных; аспект манипуляции, т.е. способы изменения состояний (модификации) данных и способы получения данных из базы данных; а также аспект целостности, т.е. описание корректных состояний базы данных. Одной из главных причин, возникающих на пути решения указанных проблем, является сложноструктурируемость лингвистического материала. Отсюда непосредственно следует сложность организации модели данных. Часть проблем была решена для двуязычного словаря [1]. Однако, как оказалось, эти решения невозможно перенести для построения трёхъязычного (русского, английского и украинского языков) электронного словаря, поскольку возникают коллизии, связанные с неадекватным отображением результатов, по причине лавинообразного нарастания переводных эквивалентов, относящихся к разным семантическим рубрикам.

В данной статье описанные трудности решаются с помощью средств теории лексикографических систем [1, 2] и алгебры конечных предикатов (АКП) [3, 4].

Кроме этого, предполагается, что электронный словарь представляет собой открытую систему. Таким образом, построенная система должна предвидеть возможность изменения нагрузки без вмешательства человека и соответственно реагировать на них. В системе необходимо предусмотреть аутентификацию пользователей, причём у разных пользователей должны быть определены разные права доступа для разного контента в разных формах.

Особенности постановки задачи исследования

Представить внутреннюю структуру двуязычного словаря относительно просто: в общих чертах, опуская особенности слов, каждому термину ставится в соответствие его переводной эквивалент. Из набора таких эквивалентов состоит словарь. Данный случай – это пример отношения один ко многим. При переходе к созданию трёхъязычного словаря, в частности, если планируется, что все три языка должны быть равноправными, появляется проблема построения связей между переводными эквивалентами. Количество связей растёт пропорционально количеству языков. Обычным решением данной проблемы является введение дополнительного уровня косвенности, что позволяет перейти от отношения многие ко многим к отношению один ко многим. Таким образом, подход к решению поставленной задачи построения трёхъязычного словаря позволит перейти от двуязычного словаря не только к трёхъязычному, но и многоязычному (для языков романо-германской группы) словарю.

Создание электронных словарей включает, как правило, следующие этапы обработки: путём сканирования и распознавания получают электронный вариант текста; далее электронный текст словаря представляется в виде массива отдельных словарных статей; дальше по формальным признакам автоматически проводится декомпозиция массива словарных статей [2].

В изначальном виде словарь был представлен в виде отсканированных и распознанных документов в формате MSWord. Использовать эти данные напрямую было невозможно из-за особенностей внутреннего представления текста в документах формата MSWord, неправильно распознанных символов, ошибок при распознавании переносов, пустых строк, разного начертания и параметров шрифтов и т.д. При этом в перечисленных выше ошибках присутствовали определённые закономерности, выявление которых позволило их исправить в автоматизированном режиме. Также все ударные буквы оказались неправильно распознанными, но единообразно, благодаря чему их удалось исправить простой заменой. Ударные буквы после парсинга1 помечаются символом «#».

Все термины из документов в MSWord формате были преобразованы в текстовый юникодный2 формат. Это стало возможным, поскольку все термины начинались с новой строки, а ошибочные символы «новой строки» были выявлены при помощи регулярных выражений. При переносе данных из формата MSWord все словосочетания были пронумерованы, поэтому поиск неправильных вхождений переводов строк был возможен при помощи регулярных выражений типа «^\d\.$» («\d\.» – служебная цифра с точкой, для нумерации перевода, добавленная в начало каждой строки). Аналогичным образом были выявлены неправильные скобки. В общем случае регулярные выражения неприменимы для определения скобок. Однако, такой подход оказался приемлем, поскольку отсутствовали сложные вложенные структуры скобок. Большинство неправильных скобок были непарными. Обычно из-за того, что разрыв слова происходил в неправильном месте при распознавании.

Пример из разбитого на переводы строк, но необработанного файла:

...
1.(моно, не)хроматѐческая
2.моно, не)- хроматѐчна аберàція)
...
1.(-витîк
2.ампåр-витîк
...

В первом случае скобка из верхней строки должна принадлежать нижней, во втором скобка – артефакт распознавания. Такие случаи необходимо было найти и соответственно можно было использовать выражения типа «\([^\)]*?$».

Самой трудоёмкой частью оказалось выявление неправильных дефисов. При помощи регулярных выражений были найдены и исправлены почти все неправильные вхождения для таких случаев. Основная масса неправильных дефисов приходилась на символ переноса слов. Такие проблемные символы были найдены при помощи регулярных выражений типа «-\s*$». Остальные дефисы пришлось исправлять в ручном режиме.

При дальнейшем парсинге были выбраны отрасль, семантика (расширенное описание), изменяемая часть и пр., что позволило заполнить внутренние структуры словаря.

Принципом построения словаря является алфавитно-гнездовой [5]. Заголовочным словом является русское слово-термин. Гнездо включает терминологические словосочетания, элементом которых является заголовочный термин. Если в заголовочную часть входит несколько однословных терминов, то для глаголов первым идёт глагол несовершенного вида, для других частей речи – наиболее общеупотребительный. Часть заглавного слова, общая для всех терминологических словосочетаний в гнезде, отделяется прямой жирной чертой (|), а в производных словах вместо неё выступает тильда (~). Если общая часть не выделена, то вместо тильды подставляется все слово. Видовые пары глаголов разделяются косой чертой (/), первым идёт глагол несовершенного вида. Терминологические словосочетания строятся таким образом, чтобы тильда была на первом месте.

Соответствующие украинские переводные эквиваленты сохраняют порядок слов русского словосочетания, английские эквиваленты сохраняют порядок слов, естественный для текстов.

Описание структуры программной системы трёхъязычного словаря

Программная система построена с использованием шаблона проектирования MVVM (Model-View-ViewModel, Модель–Представление–Модель представления). Представление – это термин в рамках шаблона проектирования MVVM, обозначающий классы, обеспечивающие работу с графическим интерфейсом пользователя. Благодаря использованию данного шаблона интерфейс пользователя оказывается отделённым от логики программы, что позволяет осуществлять независимые изменения отдельных частей программной системы.

Рассмотрим модель Trilingual Dictionary. Trilingua lDictionary – это класс, который описывает сущность трёхъязычного словаря. Он состоит из

[1] *Парсинг* (от англ. *parse*) или синтаксический анализ - процесс анализа или разбора определённого контента на составляющие с помощью роботов-парсеров (специальных программ или скриптов).

[2] *Юникод* (англ. *Unicode*) — стандарт кодирования символов, позволяющий представить знаки письменных языков.

множества терминов. Обеспечивает операции доступа к терминам, операции по добавлению/ удалению/изменению терминов. А также предоставляет операции по сохранению/загрузке словаря на/с жёсткого диска. Схема базы данных, на основе которой строится словарь, приведена на рис. 1.

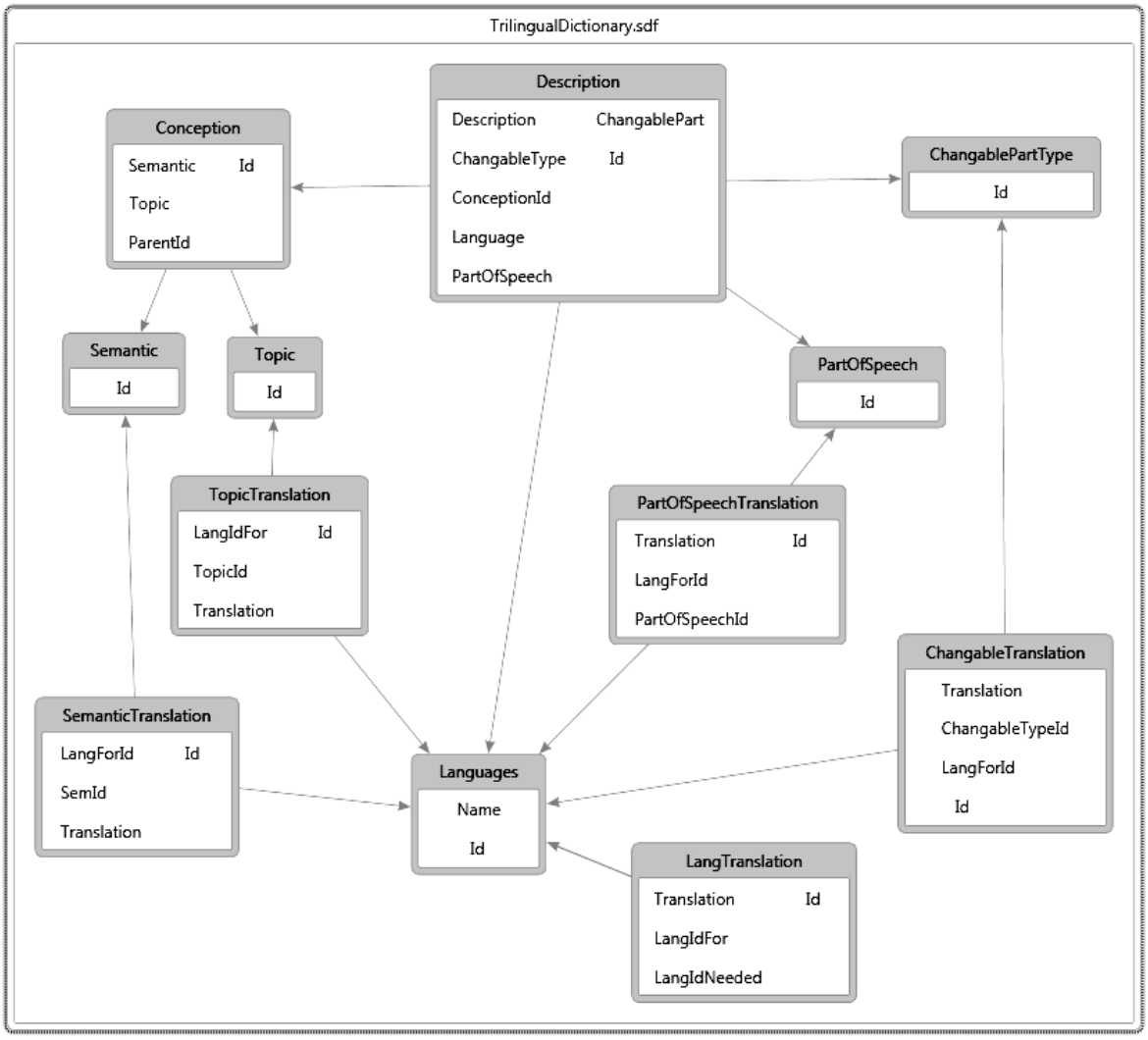

Рис. 1. Схема базы данных словаря

Класс Conception описывает термин и содержит следующие поля:

а) идентификатор термина (Id);

б) ссылку на родительский термин (ParentId);

в) раздел знаний, в котором используется термин (например, мат.- математика, рлк – радиолокация и т.д.);

г) семантическая рубрика термина, особенно актуально для омонимических терминов (например, Автозахват – Автозахоплювач (устройство); Автозахват – Автозахоплення (действие));

д) ссылку на синонимический термин;

е) множество описаний термина (представлены в классе ConceptionDescription) для разных языков.

Таким образом, обеспечиваются операции доступа к описаниям термина на разных языках, а также операции по добавлению/удалению/изменению описаний.

Класс ConceptionDescription содержит описание термина и содержит следующие поля:

а) описание самого термина;

б) ссылку на термин;

в) изменяемая часть термина для языка, используемого для описания (на данный момент родительный падеж и множественное число);

г) части речи, если это существенно;

К классам переводов относится набор классов, которые обеспечивают перевод разделов знаний, семантики, частей речи, названий языков на все доступные языки приложения.

Диаграмма классов представлена на рис. 2.

Остановимся подробнее на модели представления. Модель представления предоставляет модель данных и поведение для представления, но позволяет представлению выполнять декларативную привязку к модели представления. Модель данных представляет доступные для приложения данные, а модель представления подготавливает модель для её привязки к представлению.

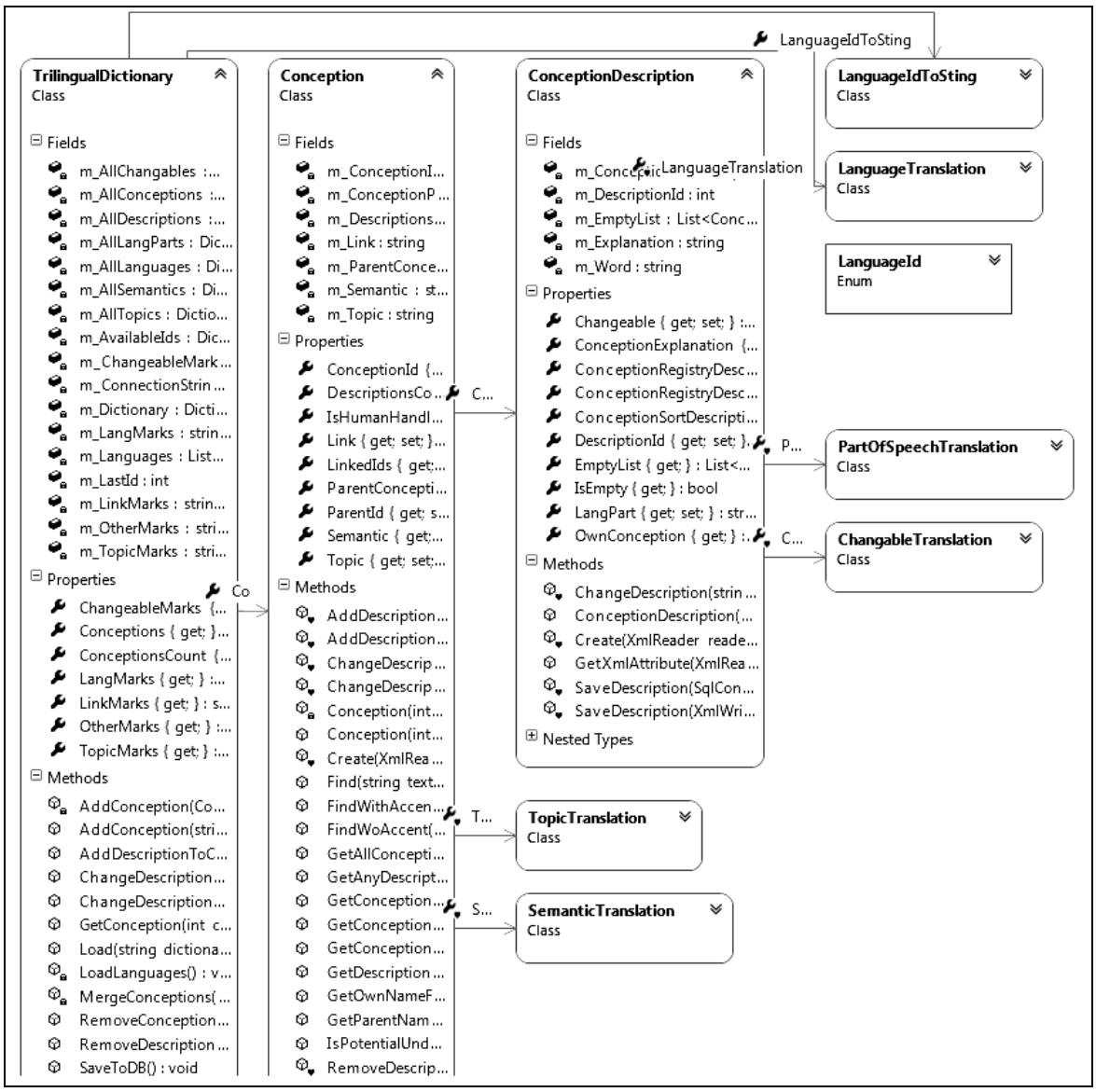

Рис. 2. Диаграмма классов ядра программной системы

Модель представления содержит классы, которые проводят адаптацию данных полученных из классов модели.

Класс TrilingualDictionaryViewModel содержит поля для отображения словаря в представлении. Он содержит следующие поля:

а) список алфавитов (класс Alphabet) для всех языков;

б) основной выбранный язык.

Класс Alphabet служит контейнером для группировки слов и словосочетаний в алфавитном порядке. Он содержит следующие поля:

а) идентификатор языка, алфавит которого хранится в объекте этого класса;

б) список букв (класс Chapter) языка.

Класс Chapter служит контейнером для группировки слов и словосочетаний в алфавитном порядке в пределах одной буквы алфавита для терминов, у которых отсутствует родительский термин (основных терминов).

Он содержит следующие поля:

а) название раздела, обычно буква алфавита;

б) список представлений описаний (класс ConceptionDescriptionViewModel) для основных терминов.

Класс *ConceptionDescriptioViewModel* служит контейнером для группировки слов и словосочетаний в алфавитном порядке для терминов, у которых существует родительский термин (дополнительных терминов). Например, «аберрация» – основной термин, «хроматическая аберрация» – дополнительный термин. Класс содержит следующие поля:

а) объект ConceptionDescription;

б) список представлений описаний (класс ConceptionDescriptionViewModel) для дополнительных терминов.

Класс ConceptionDescriptionEditViewModel предоставляет поля для изменения/добавления описаний терминов на всех доступных языках. Он содержит следующие поля и методы:

а) описание термина для выбранного языка;

б) описание раздела термина;

в) описание семантики термина;

г) описания изменяемой части описания термина;

д) части речи описания термина;

е) операции по добавлению/удалению /изменению описаний терминов.

Класс Conception View Model предзначен для представления термина с переводами на все доступные языки. Он содержит следующие поля и методы:

а) описания термина для всех доступных языков;

б) операции по выбору описаний термина для последующей модификации при помощи Conception Description Edit View Model.

Диаграмма классов представлений представлена на рис. 3.

Под представлением пониманием отображение данных словаря, полученных из модели представления.

В сущности это графический интерфейс программной системы. Представление реагирует на событие изменения значений, свойств или команд, предоставляемых моделью представления. При взаимодействии пользователя с элементами интерфейса, представление вызывает соответствующую команду, предоставленную моделью представления. Таким образом, происходит обмен-взаимодействие между интерфейсом пользователя и логикой программной системы.

Описание программной системы трёхъязычного словаря для пользователя

Программная система трёхъязычного словаря предназначена для создания, редактирования, просмотра словарных статей и их переводных эквивалентов на трёх языках.

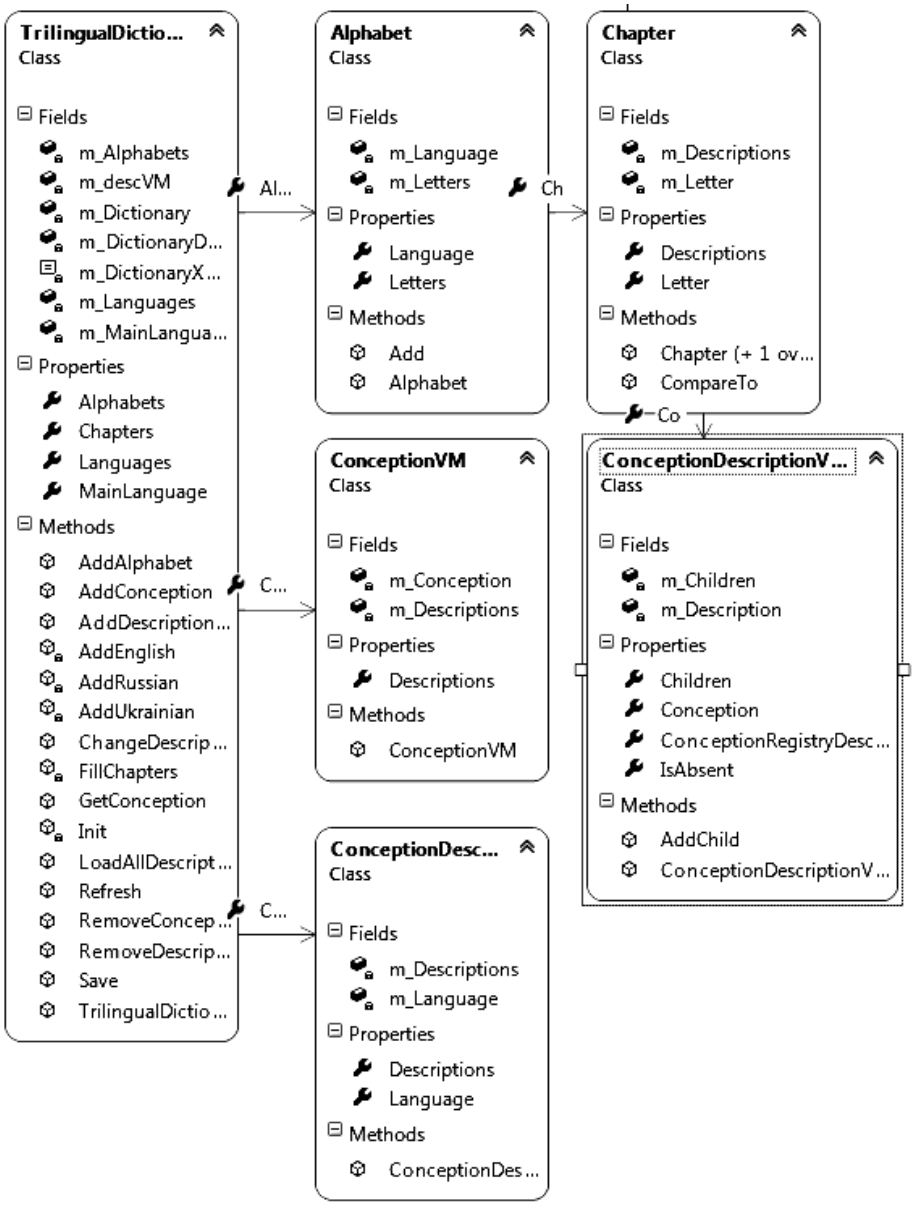

Рис. 3. Диаграмма классов представлений

Минимальные системные требования, которым она удовлетворяет, составляют:

процессор 1000 МГц;

оперативная память 1024 Мб;

не менее 100 Мб свободного места на жестком диске;

операционная система не ниже Windows XP SP3;

.Net Framework 4 Client Profile.

Интерфейс пользователя представлен на рис. 4. Левая часть окна представляет весь список терминов, из которых состоит словарь. В выпадающем меню «Основной язык» выбирается основной язык, переводные эквиваленты к словам-терминам на котором будут найдены. Они будут представлены в списке в левой панели. В окне «Поиск» можно ввести часть слова, которое необходимо найти. Знак «#» в словах обозначает ударение, поиск может осуществляться как с учётом ударений, так и без их учёта. После нажатие кнопки «Найти» производится поиск термина в левой панели.

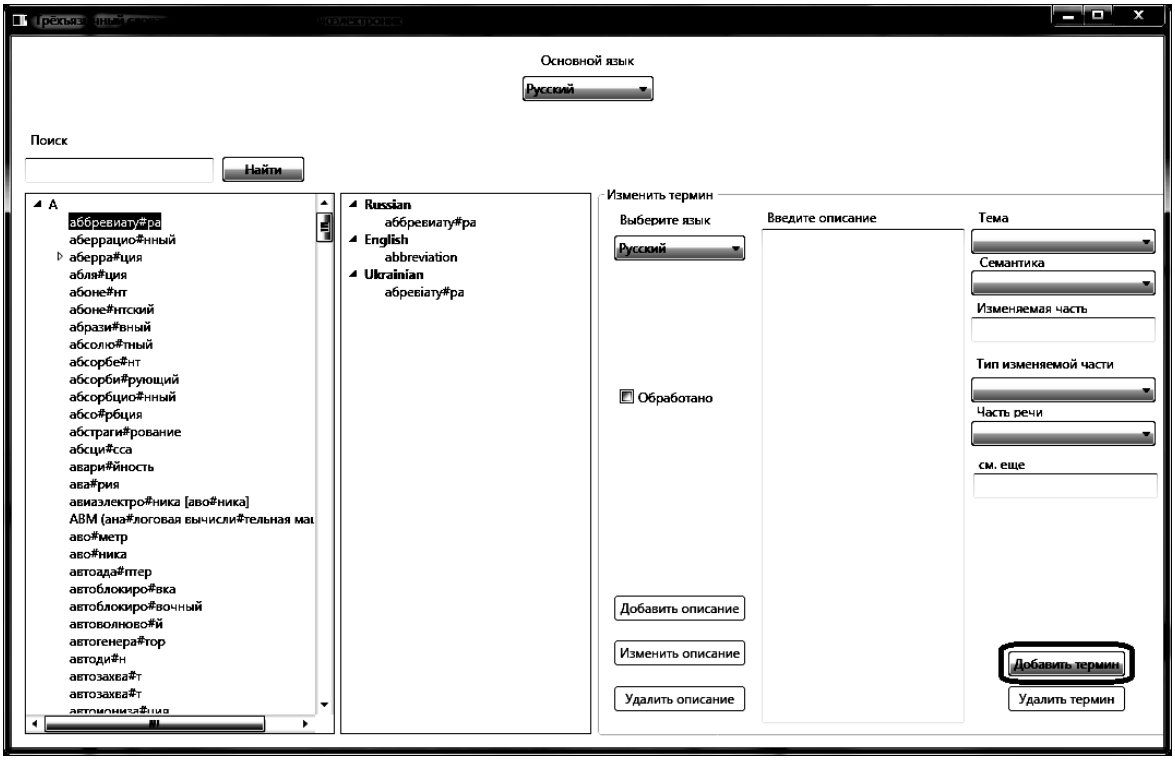

Рис. 4. Главное окно словаря

Правая панель предназначена для редактирования терминов. Для редактирования описания существующего термина, он должен быть выбран в левой панели. Для создания/изменения/описания термина в выпадающем меню «Выберите язык» необходимо выбрать язык, для которого осуществляется редактирование.

В поле «Введите описание» отображается описание термина, который выбран в левой панели на выбранном языке. В этом поле необходимо внести нужные изменения в описание выбранного термина.

В поле «Тема» вводится условное обозначение отрасли термина (например, мат., физ. и т.д.)

В поле «Семантика» вводится расширенное описание (например, слово «акт» может иметь два значения «действие» и «документ»)

В поле «Изменяемая часть» вводится либо слово целиком, если необходимо показать переход ударения или изменение фонем (например, для слова «вісь» в этом поле будет «осі»), либо окончание (или какая-нибудь другая изменяемая часть) слова (например, для слова «дейтрон» в этом поле будет «-на», а для слова «нелінійність» – «-ності»).

В поле «Тип изменяемой части» вводится тип изменяемой части (например, родительный падеж «род.», множественное число «мн.»).

В поле «Часть речи» вводится часть речи, если это необходимо. При необходимости внести изменения в описание существующего термина, после завершения редактирования описания для применения изменений необходимо нажать одну из кнопок слева. Кнопки редактирования будут активными в зависимости от того, существовало ли описание термина раньше, до его редактирования.

Кнопка «Добавить описание» добавляет описание для выбранного языка, выбранного термина. Кнопка активна, если описание на выбранном языке отсутствует.

Кнопка «Изменить описание» изменяет описание для выбранного языка, выбранного термина. Кнопка активна, если описание на выбранном языке уже существует.

Кнопка «Удалить описание» удаляет описание для выбранного языка, выбранного термина. Кнопка активна, если описание на выбранном языке уже существует. Если необходимо создать термин, то после завершения редактирования описания для применения изменений необходимо нажать кнопку справа «Добавить термин». После этого откроется окно добавления термина (рис. 5). В словарь будет добавлен новый термин с описанием на языке, на котором выполнялось редактирование. Добавление описаний для других языков осуществляется по процедуре описанной выше.

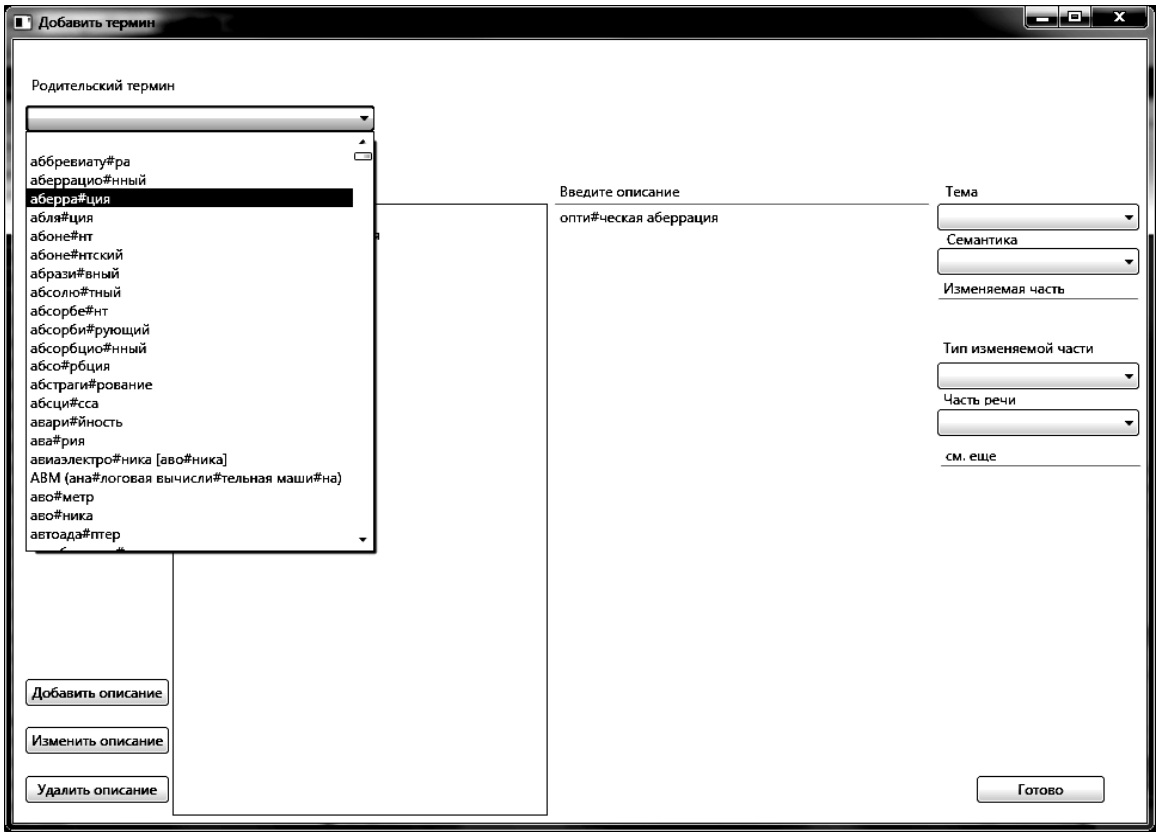

Рис. 5. Добавление описаний для нового термина и выбор родительского термина

При нажатии на кнопку «Удалить термин» термин удаляется из словаря целиком, со всеми описаниями. Кнопка активна, если какой-либо термин выбран в левой панели. Для того, чтобы посмотреть описание термина, необходимо выбрать его в левой панели. В средней панели будут отображены описания термина на всех доступных языках в виде древообразной структуры, сгруппированной по языкам. Поле «Изменить термин» будет заполнено в зависимости от выбранного описания в средней панели.

Примеры использования программной системы трёхъязычного словаря

Программная система трёхъязычного словаря предназначена для создания, редактирования, просмотра словарных статей и их переводных эквивалентов на трёх языках.

Создание термина. Для создания термина пользователь должен нажать на кнопку «Добавить термин», на рис. 4 обведена красным цветом. После этого откроется окно добавления термина. Необходимо ввести описание термина на выбранных языках.

На рис. 5 показано добавление описания термина для английского языка. Для русского и украинского языков описания уже добавлены.

При необходимости в выпадающем меню можно выбрать родительский термин, в данном примере им является термин «аберрация».

После того, как все данные введены, необходимо нажать кнопку «Готово» и новый термин будет добавлен в базу данных. Редактирование терминов. Для редактирования предаврительно добавленных терминов необходимо выбрать нужный термин в списке на левой панели. Выбранный термин с переводами будет отображен на средней панели (см. рис. 4). После ввода описания на нужном языке, надо нажать кнопку «Добавить описание», и оно будет добавлено к термину.Для изменения термина, необходимо на средней панели выбрать его описание на нужном языке. Изменить описание на правой панели и нажать кнопку «Изменить описание». Изменения отобразятся на средней панели.

Переключение основного языка словаря. При необходимости можно переключить основной язык в выпадающем меню в верху экрана.

Заключение

Таким образом, была построена программная система русско-украинско-английского терминологического словаря. При её построении удалось уйти от некоторых проблем, которые возникали в ранее построенных словарях [1, 6]. Так удалось автоматизировать корректировку входных данных, применив регулярные выражения. Кроме этого, применение математического аппарата теории лексикографических систем и алгебры конечных предикатов [6, 7] позволило уйти от избыточности в решениях. Для формализации языковой информации использовалось понятие семантического состояния языковой единицы [6]. А для адекватного построения модели данных и модели представления необходимо было формально описать задачу средствами алгебры конечных предикатов, такими как линейные логические преобразования, а также методы построения реляционных сетей [8 – 10]. Очевидно, что поход к построению трёхъязычного словаря можно также применить и для словарей с большим количеством языков.

В статье подробно описана структура программной системы трёхъязычного словаря. Приведена соответствующая схема базы данных, описаны классы сущностей и классы модели представления. Также в статье подробно представлено пользовательское описание программной системы.

Разработанная программная система позволяет редактировать, наполнять и создавать новые тематические переводные электронные словари. Словарь удобен как для администратора, так и для непосредственного пользователя. Несомненным преимуществом системы есть то, что каждый из предложенных языков равноправен изначально, пользователь сам выбирает главный язык в каждом конкретном случае.

Список литературы

1. Широков В.А. Комп'ютерна лексикографія / В.А. Широков. – К.: науково виробниче підприємство «Видавництво «Наукова думка» НАН України», 2011. – 352 с.

2. Рабулець О.Г. Дієслово в лексикографічній системі / О.Г. Рабулець, В.А. Широков, К.М. Якименко. – К.: Довіра, 2004. – 259 с.

3. Бондаренко М.Ф. Теория интеллекта: учеб. / М.Ф. Бондаренко, Ю.П. Шабанов-Кушнаренко. – Харьков: Изд-во СМИТ, 2006. – 571 с.

4. Бондаренко М.Ф. Мозгоподобные структуры: справочное пособие. Т. 1 / М.Ф. Бондаренко, Ю.П. Шабанов-Кушнаренко. – К.: Наукова думка, 2011. – 460 с.

5. Остапова И.В. Лексикографическая структура этимологических словарей и их представление в цифровой среде / И.В. Остапова // Прикладная лингвистика и лингвистические технологи. – 2007. – С. 236-245.

6. Широков В.А. Елементи лексикографії / В.А. Широков. – К.: Видавництво "Довіра", 2005. – 303 с.

7. Вечирская И.Д. Разработка трехязычного терминологического словаря на основе алгебры конечных предикатов / И.Д. Вечирская // Бионика интеллекта: науч.-техн. журнал. – 2011. – № 2(76). – С. 109-113.

8. Четвериков Г.Г. Формалізація принципів побудови універсальних k-значних структур мовних систем штучного інтелекту / Г.Г. Четвериков // Доповіді НАН України. – 2001.– №1 (41). – С. 76-79.

9. Вечірська І.Д. Дослідження розмірності предметного простору в задачах моделювання об'єктів у вигляді реляційних мереж / І.Д. Вечірська // Біоніка інтелекту: наук.-техн. журнал. – 2009. – № 2 (71). – С. 31-35.

10. Бондаренко М.Ф. Концепції уніфікації інформаційно-інтелектуальних технологій в системах мовлення / М.Ф. Бондаренко, З.Д. Коноплянко, Г.Г. Четвериков // Бионика интеллекта. – 2011. – № 3 (77). – С. 150-156.

Поступила в редколлегию 5.05.2016

Рецензент: д-р техн. наук, проф. С.Ю. Шабанов-Кушнаренко, Харьковский национальный университет радиоэлектроники, Харьков.

РОЗРОБКА ПРОГРАМНОЇ СИСТЕМИ ЕЛЕКТРОННОГО ТРИМОВНОГО СЛОВНИКА

Г.Г. Четвериков, І.Д. Вечірська, О.С. Пузік

Статтю присвячено розробці програмної системи російсько-українсько-англійського термінологічного словника. Вхідними даними являються відсканованні та розпізнані документи у форматі MSWord. Проаналізовано помилки, що виникають при синтаксичному аналізі вхідних даних, та показано шляхи їх усунення за допомогою регулярних виразів. Наведено та детально описано схему лексикографічної бази даних словника, описано класи моделі даних та класи моделі представлення системи. Окрім того, представлено детальний опис програмної системи щодо користувача, а також визначено перспективи використання як самого словника, так і засобів його побудови. Програмну систему побудовано із використанням шаблону проектування Model-View-ViewModel. Використання цього шаблону дозволило відділити інтерфейс користувача від логіки програми. Це дозволяє виконувати незалежні зміни окремих частин програмної системи. Розроблена програмна система дозволяє редагувати, наповнювати і таким чином створювати нові тематичні перекладні електронні словники. Перевагаю системи є рівноправність мов, у кожному окремому випадку користувач визначає сам, яка з мов буде головною.

Ключевые слова: *алгебра скінченних предикатів, база даних, лексикографія, лексична одиниця, шаблон MVVM, парсинг, програмна система.*

DEVELOPMENT OF SOFTWARE ELECTRONIC TRILINGUAL DICTIONARY

G.G. Chetverikov, I.D. Vechirska, O.S. Puzik

The article is devoted to development of a software system of the Russian-Ukrainian-English dictionary of terminology. Scanned and recognized documents in MSWord format are input data for the dictionary. Issues appeared during parsing of input data have been analyzed and identified ways to resolve those using regular expressions. The article describes scheme of lexicographical database of the dictionary, classes of models, views and view models. In addition, detailed description of the software system from a user perspective has been presented, and emphasized perspectives of usage of the dictionary and the methods being used during development. The software system is built using design pattern Model-View-View Model. Through the use of this pattern business logic is separated from user interface, thus changes made in different parts of the software may be independent. The developed software system allows to edit, to fill, and thus to create new thematic transferable electronic dictionaries. The advantage of the system is the equality of languages, i.e. in each case user can assign a language that is a major.

Keywords*: algebra of finite predicates, database, lexicography, lexical unit, MVVM template, parsing, software system.*