# Задача построения трёхъязычного словаря.

Представить внутреннюю структуру двуязычного словаря относительно просто: в общих чертах, опуская особенности слов, каждому термину ставится в соответствие его переводной эквивалент и из набора таких эквивалентов состоит словарь. Данный случай – это пример отношения один к одному.

При переходе к созданию трёхъязычному словарю, тем более, если планируются, что все три языка должны быть равноправными, появляется проблема построения связей между переводными эквивалентами. Количество связей растет пропорционально количеству языков. Данный случай – это пример отношения многие ко многим. Обычным решением данной проблемы является введение дополнительного уровня косвенности, что позволяет перейти от отношения 1:1 к отношению 1:М. Таким образом данный подход позволяет перейти от двуязычного к трёхъязычному и многоязычному словарям.

# Описание структуры программной системы трёхъязычного словаря

Программная система построена с использованием шаблона проектирования MVVM (Model-View-View Model, Модель-Представление-Модель представления). Благодаря использованию данного шаблона интерфейс пользователя оказывается отделен от логики программы, что позволяет независимое изменение отдельных частей программной системы.

## Модель

### TrilingualDictionary

TrilingualDictionary – это класс, который описывает сущность трёхъязычного словаря. Он который состоит из коллекции терминов. Обеспечивает операции доступа к терминам, операции по добавлению/удалению/изменению терминов. Предоставляет операции по сохранению/загрузке словаря на/с жесткого диска.

Схема базы данных на основе которой строится словарь приведена на рисунке 1.

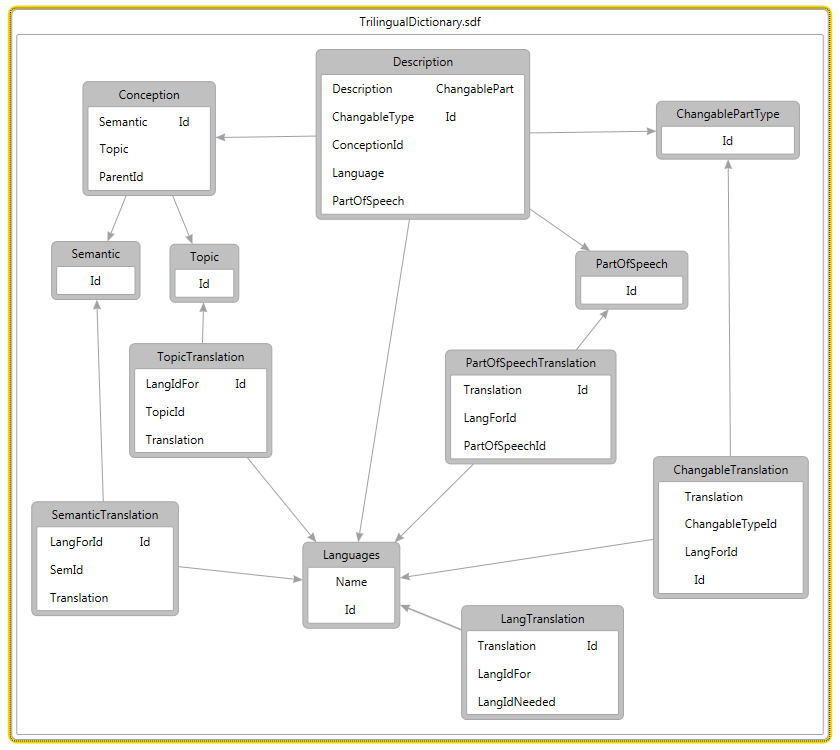


Рисунок 1 – Схема базы данных

### Conception

Этот класс описывает термин. Он содержит следующие поля:

а) идентификатор термина;

б) ссылку на родительский термин;

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

г) семантику термина;

д) ссылку на синонимический термин;

е) коллекцию описаний термина (описаны в классе ConceptionDescription) для разных языков;

Обеспечивает операции доступа к описаниям термина на разных языках, операции по добавлению/удалению/изменению описаний.

ConceptionDescription

Этот класс содержит описание термина. Он 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/применимо;

### Классы переводов

Набор классов, которые обеспечивают перевод разделов знаний, семантики, частей речи, названий языков на все доступные языки приложения.

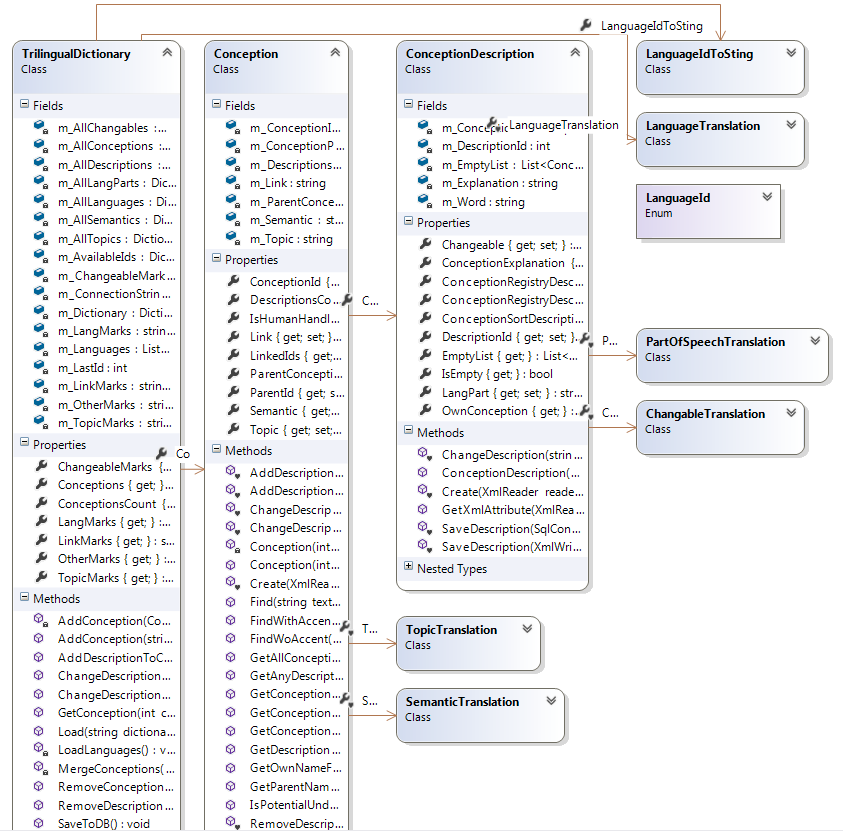
Диаграмма классов представлена на рисунке 2. 

Рисунок 2 – Диаграмма классов ядра

## Модель представления

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

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

### TrilingualDictionaryViewModel

Этот класс предоставляет поля для отображения словаря в представлении. Он содержит следующие поля:

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

б) основной выбранный язык.

### Alphabet

Этот класс 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.

### Chapter

Этот класс 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.

### ConceptionDescriptioViewModel

Этот класс 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. Он содержит следующие поля:

а) объект ConceptionDescription;

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

### ConceptionDescriptionEditViewModel

Этот класс 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.

### ConceptionViewModel

Этот класс 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 ConceptionDescriptionEditViewModel.

Диаграмма классов представлений представлена на рисунке 3.

## Представление

Отображает данные словаря полученные из модели представления. По сути 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.

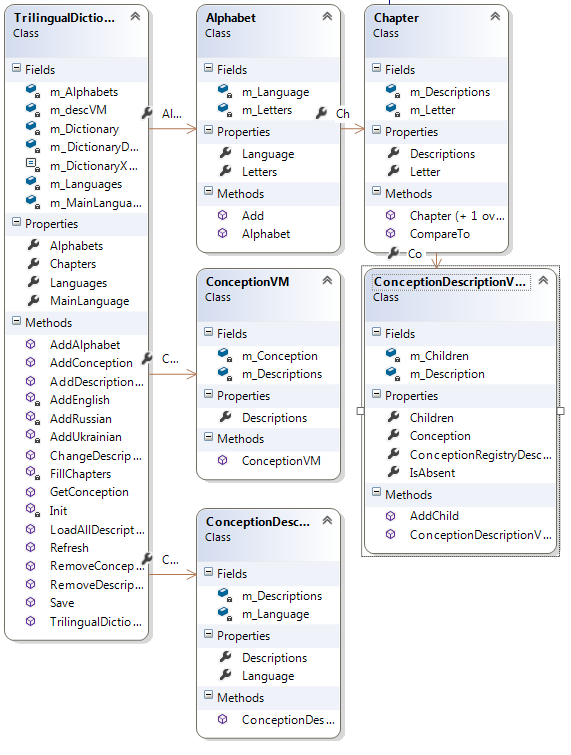


Рисунок 3 – Диаграмма классов представлений

# Пользовательское описание программной системы трёхъязычного словаря

## Общие сведения

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

Минимальные системные требования:

* процессор 1000 МГц;
* оперативная память 512 Мб;
* не менее 100 Мб свободного места на жестком диске;
* операционная система не ниже Windows XP SP3;
* .Net Framework 4 Client Profile;

## Интерфейс пользователя

Интерфейс пользователя представлен на рисунке 4.

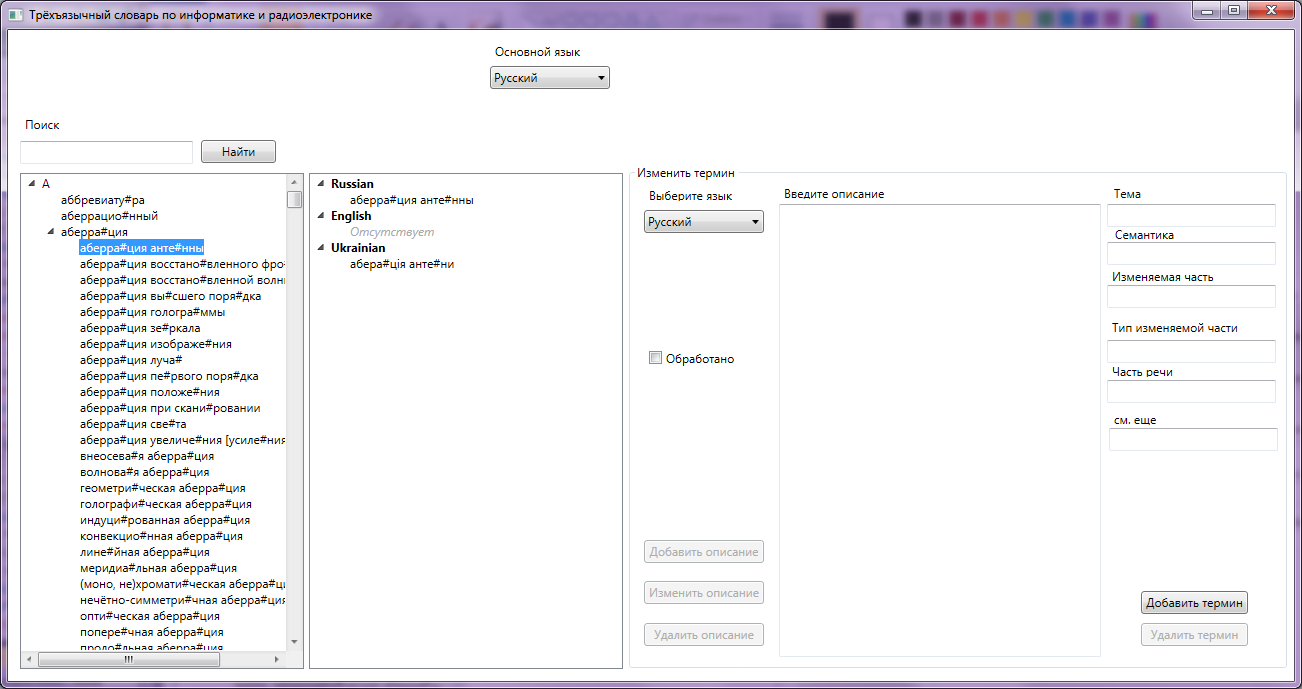


Рисунок 4 – Главное окно словаря

Левая часть окна представляет весь список терминов, из которых состоит словарь. В выпадающем меню «Основной язык» выбирается основной язык, на котором будут представлены термины 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 учетом ударений, так и без их учета. После нажатие кнопки «Найти» производится поиск термина в левой панели.

Правая панель состоит из двух частей. Верхняя часть предназначена для редактирования терминов. Нижняя часть предназначена для отображения полной информации о термине.

### Редактирование терминов

Для редактирования описания существующего термина, он должен быть выбран в левой панели.

#### Создание/изменение 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 «вiсь» в этом поле будет «осi»). Либо окончание (либо какая-либо другая изменяемая часть) слова (например для слова «дейтрон» в этом поле будет «-на», а для слова «нелінійність» - «-ностi»).

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 существенно/применимо

В поле «см. ещё» вводится отсылка к другому слову в статье.

#### Изменение описания существующего термина

Если необходимо изменить существующий термин, то после завершения редактирования описания для применения изменений необходимо нажать одну из кнопок слева. В зависимости от того существовало ли описание до его редактирования будут активными кнопки редактирования.

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

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

«Удалить описание» - удаляет описание для выбранного языка, выбранного термина. Кнопка активна, если описание на выбранном языке уже существует.

#### Добавление/удаление термина

Если необходимо создать термин, то после завершения редактирования описания для применения изменений необходимо нажать кнопку справа «Добавить термин». В словарь будет добавлен новый термин с описанием на языке, на котором выполнялось редактирование. Добавление описаний для других языков осуществляется по процедуре описанной выше.

При нажатии на кнопку «Удалить термин», термин удаляется из словаря целиком, со всеми описаниями. Кнопка активна, если какой-либо термин выбран в левой панели.

### Описание терминов

В левой панели необходимо выбрать термин, для которого надо получить описания. 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.

# Особенности парсинга отсканированного и распознанного текста

В изначальном, «сыром» виде словарь был представлен в виде отсканированных и распознанных документов в формате MSWord. Использовать эти данные напрямую было невозможно из-за неправильно распознанных символов, ошибок при распознавании переносов, пустых строк, жирного шрифта. При этом в перечисленных выше ошибках присутствовала определённая закономерность, что позволило их исправить в автоматизированном режиме.

Все термины из документов в MSWord формате были преобразованы в текстовый юникодный формат. Это стало возможным, поскольку все термины начинались с новой строки, а ошибочные символы «новой строки» были выявлены при помощи регулярных выражений.

Все ударные буквы оказались неправильно распознаны, но единообразно, благодаря чему их удалось исправить простой заменой. Ударные буквы после парсинга помечаются символом «#».

Аналогичным образом были выявлены неправильные скобки. В общем случае регулярные выражения неприменимы для определения скобок, но в данном конкретном случае это подошло, поскольку отсутствовали более-менее сложные вложенные структуры скобок.

Самой трудоемкой частью оказалось выявление неправильных дефисов. При помощи регулярных выражений были найдены и исправлены все неправильные вхождения дефисов.

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