



**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ Информатика, искусственный интеллект и системы управления

КАФЕДРА Системы обработки информации и управления

**Методические указания к лабораторным работам
по курсу «Технологии разработки программного обеспечения»**

**Лабораторная работа №6
«Прямое и обратное проектирование»**

Виноградова М.В., Королев С.В., Волков А.С.

Под редакцией к.т.н. доц. Виноградовой М.В.

Москва, 2022 г.

ОГЛАВЛЕНИЕ

| | |
|--|-----------|
| 1. ЗАДАНИЕ..... | 3 |
| 1.1. Цель работы..... | 3 |
| 1.2. Средства выполнения..... | 3 |
| 1.3. Продолжительность работы..... | 3 |
| 1.4. Пункты задания для выполнения..... | 3 |
| 1.5. Содержание отчета | 4 |
| 2. ВЫПОЛНЕНИЕ РАБОТЫ | 5 |
| 2.1. Обозначение профиля ЯП для проекта..... | 5 |
| 2.1.1. Установка ЯП для проекта или класса | 5 |
| 2.2. Отображение типов данных и стереотипов..... | 6 |
| 2.3. Трассировка классов | 8 |
| 2.3.1. Трассировка управляющих классов..... | 8 |
| 2.3.2. Трассировка классов сущностей | 8 |
| 2.3.3. Трассировка граничных классов | 10 |
| 2.4. Прямое и обратное проектирование | 10 |
| 2.5. Трассировка пакетов в подсистемы | 14 |
| 2.6. Создание инфологической модели | 15 |
| 2.6.1. Создание вручную | 15 |
| 2.6.2. Генерация из диаграммы классов модели требований | 17 |
| 2.6.3. Генерация из даталогической модели | 19 |
| 2.7. Создание даталогической модели | 20 |
| 2.7.1. Генерация из инфологической модели..... | 20 |
| 2.7.2. Создание вручную | 22 |
| 2.8. Генерация DDL-сценария | 24 |
| 2.9. Генерация документации..... | 25 |
| 2.10. Проверка целостности данных проекта..... | 26 |
| 2.11. Статистика проекта | 28 |
| 2.12. Пример проектирования базы данных | 29 |
| 2.12.1. Шаблон базы данных | 29 |
| 2.12.2. Генерация DDL-сценария | 31 |
| 2.12.3. Генерация документации..... | 32 |
| 3. КОНТРОЛЬНЫЕ ВОПРОСЫ | 34 |
| 4. СПИСОК ИСТОЧНИКОВ..... | 35 |
| 5. ПРИЛОЖЕНИЕ 1 Описание пунктов генерации документации | 36 |

1. ЗАДАНИЕ

Лабораторная работа №6 «Прямое и обратное проектирование» по курсу «Технологии разработки программного обеспечения».

1.1. Цель работы

- Изучить унифицированный процесс разработки (RUP);
- Приобрести умения построения модели проектирования и процессов прямого и обратного проектирования;
- Получить навыки построения модели проектирования и процессов прямого и обратного проектирования в среде Sparx Enterprise Architect.

1.2. Средства выполнения

- Sparx Enterprise Architect [1].

1.3. Продолжительность работы

Время выполнения лабораторной работы 4 часа.

1.4. Пункты задания для выполнения

1. Открыть в среде Enterprise Architect проект, созданный ранее.
2. Установить для проекта профиль языка программирования, который будет использоваться при обозначении типов данных и генерации кода.
3. Создать диаграмму трассировки граничных классов в классы форм.
4. Создать диаграмму трассировки управляющих классов в классы проектирования [2]. Создать подсистему (package) и переместить туда полученные классы.
5. Выполнить прямое и обратное проектирование package подсистемы (сгенерировать код, изменить код, синхронизировать).
6. Создать диаграмму трассировки пакетов анализа в подсистемы.
7. Создать инфологическую модель.
8. Создать диаграмму трассировки классов сущностей в сущности инфологической модели.

9. Сгенерировать даталогическую модель на основе инфологической модели.
10. Сгенерировать DDL-сценарий на основе даталогической модели.
11. Сгенерировать документацию проекта.
12. Выполнить проверку целостности проекта.
13. Получить краткую статистику проекта.
14. Построить и изучить шаблон проекта базы данных (DDL, документация).

1.5. Содержание отчета

- Титульный лист;
- Цель работы;
- Задание;
- Диаграммы трассировки классов анализа в классы проектирования;
- Описание классов проектирования, их атрибутов, операций и их параметров;
- Диаграмма трассировки пакетов в подсистемы;
- Инфологическая и даталогическая модель (диаграммы);
- DDL-сценарий;
- Пример документации проекта (скриншот);
- Результат проверки целостности проекта;
- Краткая статистика проекта;
- Шаблон проекта базы данных (содержание);
- Вывод;
- Список используемой литературы.

2. ВЫПОЛНЕНИЕ РАБОТЫ

2.1. Обозначение профиля ЯП для проекта

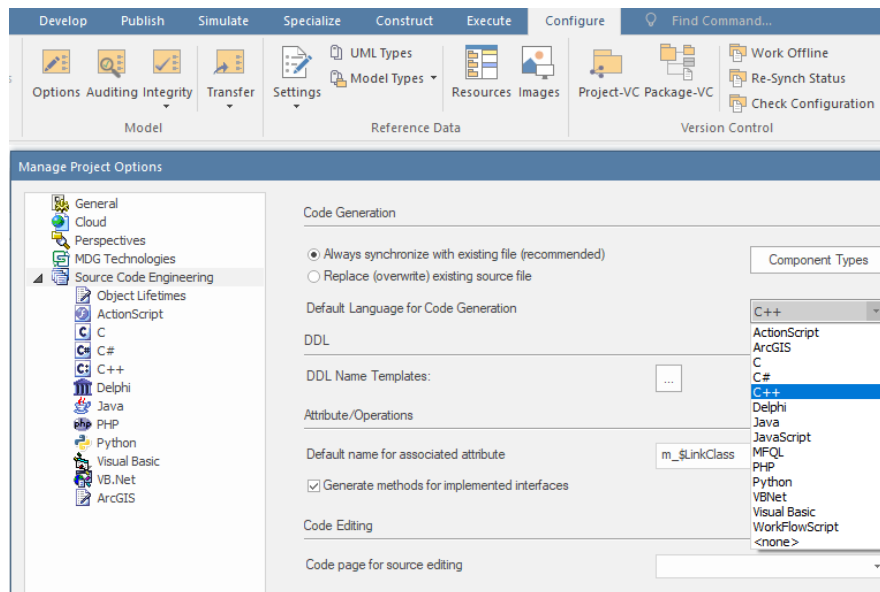
Определение языка программирования (ЯП), который будет использоваться при разработке, необходимо для того, чтобы в дальнейшем на диаграммах использовались те типы данных, которые поддерживаются данным языком, а также чтобы на требуемом языке выполнить генерацию кода по построенной диаграмме.

2.1.1. Установка ЯП для проекта или класса

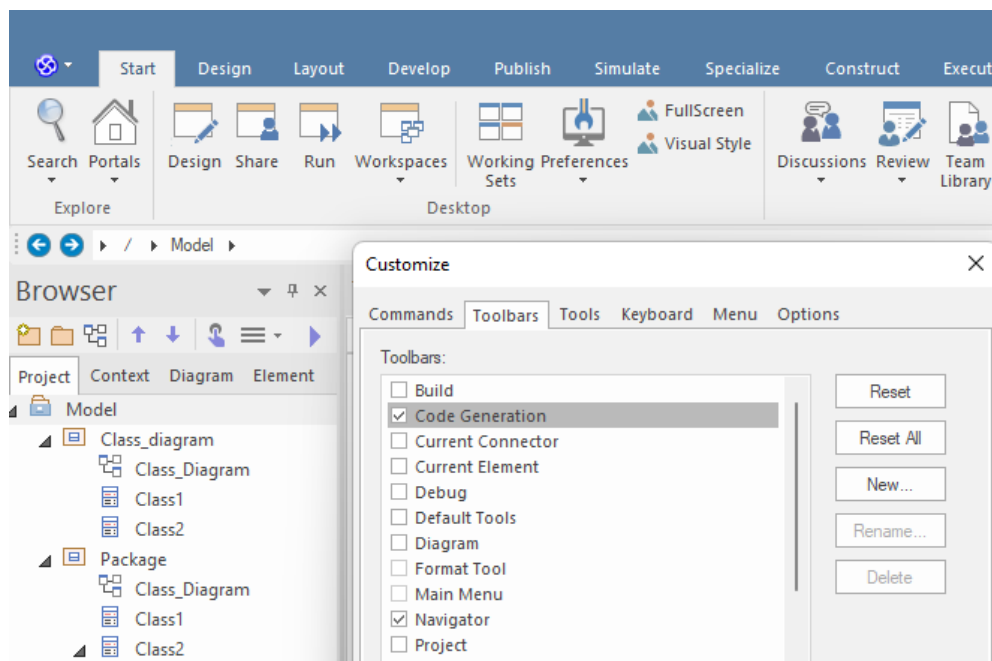
Определить ЯП можно:

- **На уровне проекта:**

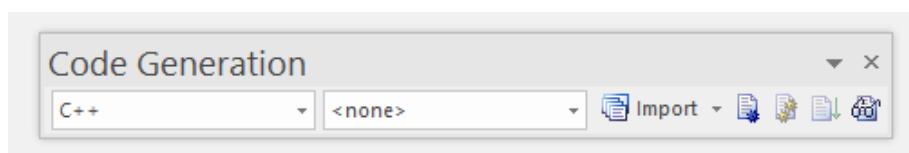
Для этого необходимо перейти в пункт `Configure>Model>Options>Source Code Engineering`, выбрать ЯП из выпадающего списка в `Default Language for Code Generation`. Для примера будем использовать язык C++, он наиболее подробно описан в среде Enterprise Architect.



Альтернативно можно включить плавающее окно выбора ЯП. Для этого перейти `Start>Desktop>Preferences>Other Options>Toolbars` и выбрать `Code Generation`.



После этого появится небольшое окно, в котором можно управлять выбором ЯП и автоматической генерацией кода.

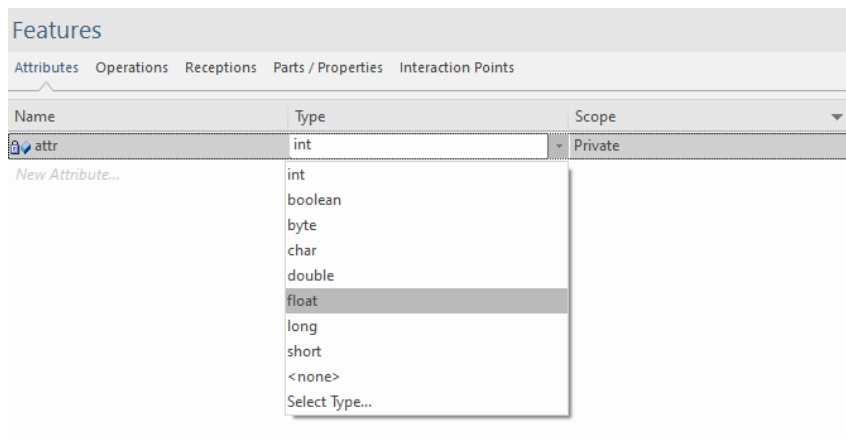


- **На уровне класса:**

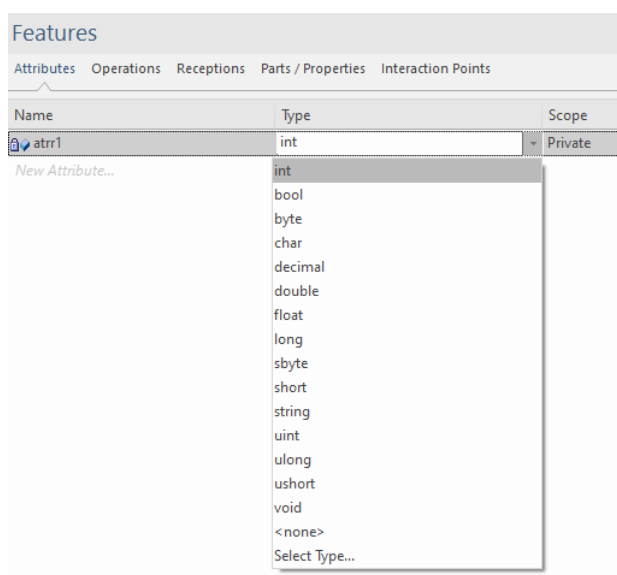
В таком случае можно выбирать ЯП для каждого конкретного класса на диаграмме. Для этого кликнуть ПКМ по существующему классу и выбрать пункт Properties>Properties... На вкладке General справа можно выбрать ЯП, который будет использоваться при описании класса.

2.2. Отображение типов данных и стереотипов

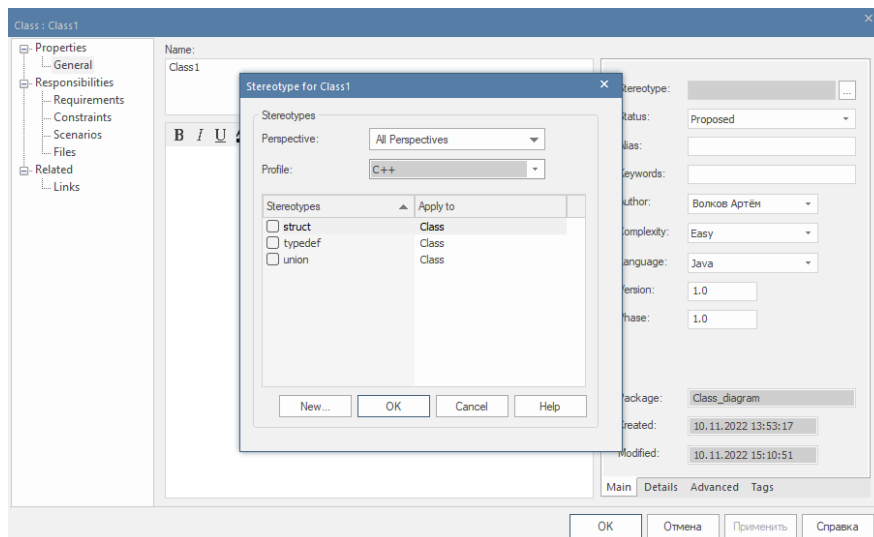
После выбора ЯП можно заметить, что набор типов данных для атрибутов и методов соответствует данному ЯП. Рисунок ниже демонстрирует выбор типов данных для класса на языке C++.



Если выбрать другой ЯП, то список возможных типов данных также поменяется. Ниже представлен список типов данных для класса на языке C#. Обратите внимание, на новые типы данных.



Также для классов можно обозначить стереотипы, которые поддерживает определенный ЯП. Для этого кликнуть ПКМ по существующему классу и выбрать пункт Properties>Properties... На вкладке General справа кликнуть по «...» по пункту Stereotype. В открывшемся окне можно выбрать ЯП из выпадающего списка Profile. Для выбранного ЯП отобразится список поддерживаемых стереотипов.



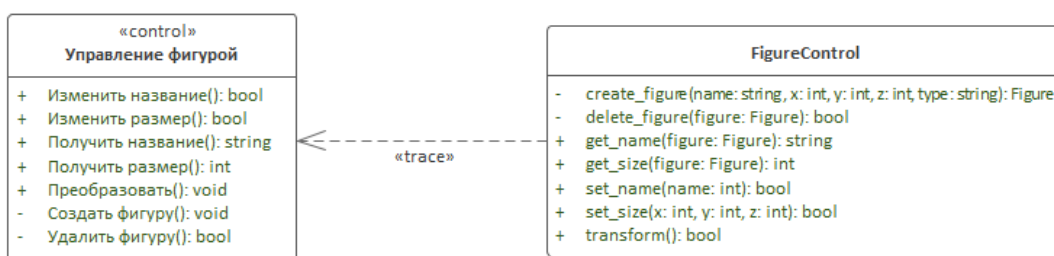
2.3. Трассировка классов

Классы анализа трассируются в классы проектирования. При этом классы проектирования описываются с учетом особенностей языка программирования и разрабатываемого проекта:

- Точное указание типов данных;
- Соблюдение стереотипов;
- Указание параметров методов;
- Точное указание доступности (score) методов и атрибутов;
- Написание названий элементов на латинице предпочтительно с соблюдением правил написания названий элементов (camelCase, snake_case и т.п.)

2.3.1. Трассировка управляющих классов

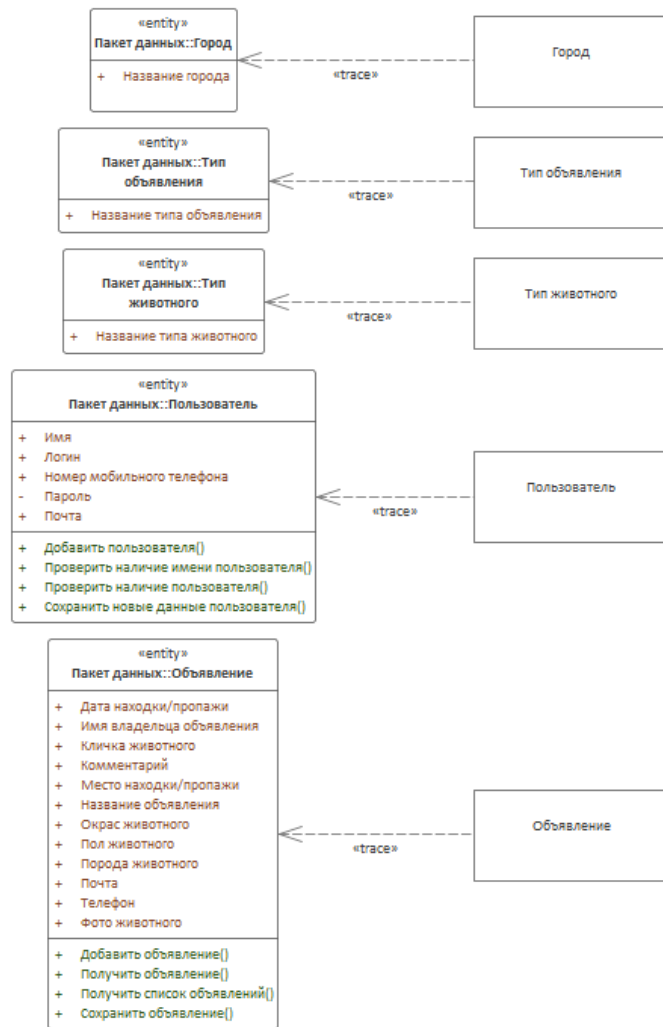
Ниже представлен пример трассировки (стереотип связи «trace») управляющего класса анализа в класс проектирования.



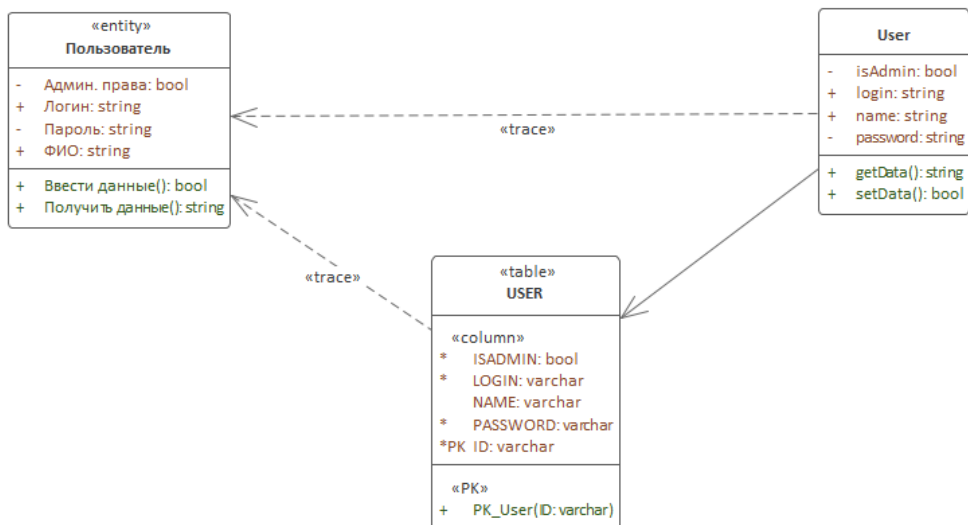
2.3.2. Трассировка классов сущностей

В рамках данной лабораторной работы классы сущности трассируются в сущности инфологической модели. Описание построения инфологической модели

приводится в следующем разделе. Ниже представлена трассировка классов сущностей в заранее полученные сущности инфологической модели.



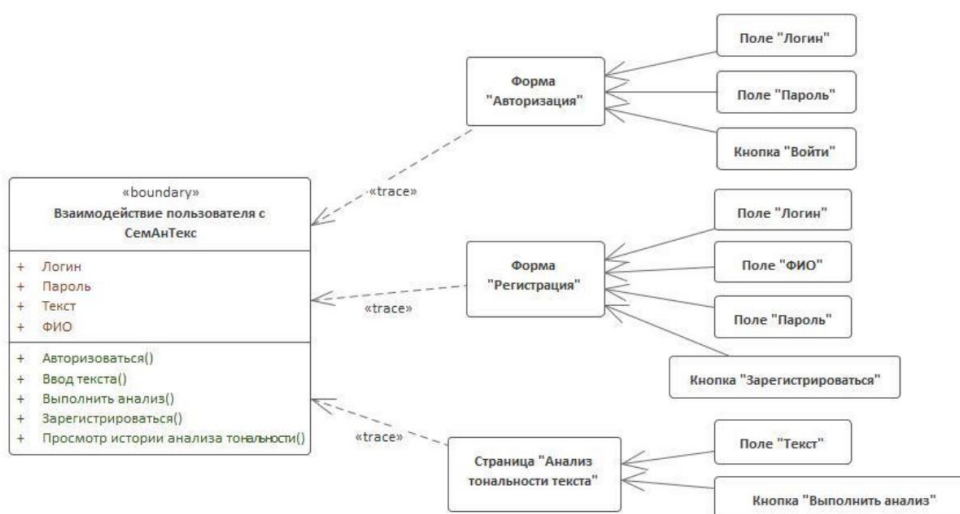
Также классы сущностей могут трассироваться классы проектирования с учетом создания не только отдельного класса, но и таблицы в БД.



Создание таблицы на диаграмме классов происходит следующим образом: перенести на диаграмму пустой класс, выбрать для него стереотип «table», применить изменения (отменить выделение и заново выделить класс). Теперь вместо возможности добавлять атрибуты и методы у этого класса будет доступна только возможность добавления колонок (Column).

2.3.3. Трассировка граничных классов

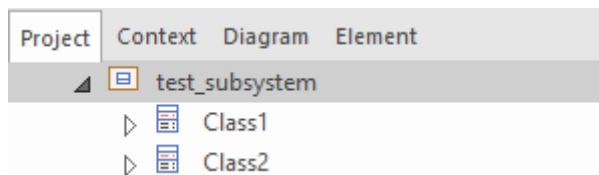
Граничные классы трассируются в формы пользовательского интерфейса и элементы этих форм, каждая из которых представлены на диаграмме отдельными классами.



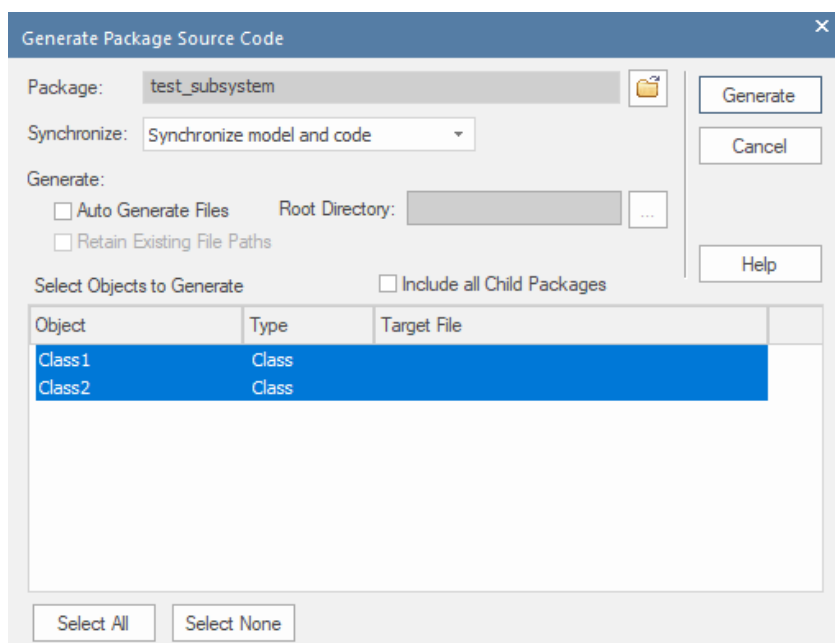
2.4. Прямое и обратное проектирование

В данном разделе под прямым проектированием имеется в виду генерация кода из содержимого диаграммы, а обратное проектирование – обновление диаграммы в соответствии с изменениями, внесенными в код.

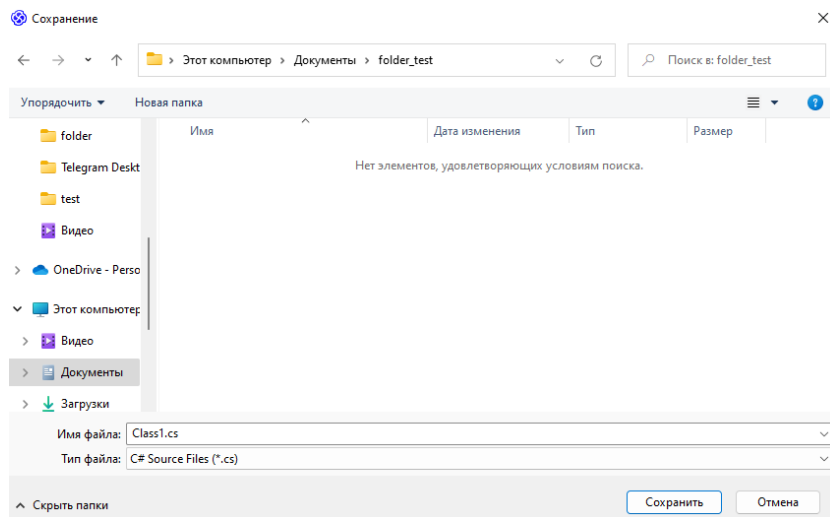
Реализуем прямое и обратное проектирование на примере двух классов: Class1 и Class2. Class1 связан с Class2 отношением наследования (Generalization). Данные классы были созданы на отдельной диаграмме, а затем были перенесены в заранее созданный пакет test_subsystem.



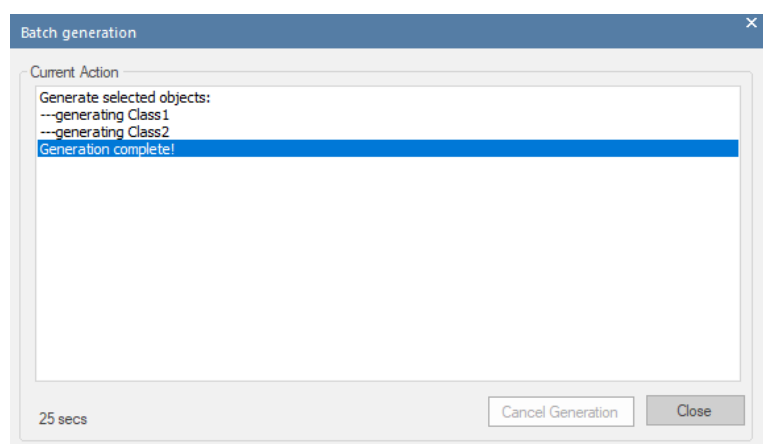
Выполним генерацию кода по данным классам – **прямое проектирование**. Для этого необходимо перейти в пункт меню **Develop>Source Code>Generate>Generate All**.



Далее необходимо выбрать папку, в которую будут сохраняться программные коды.



В результате будет выведен лог-файл результатов генерации.



Откроем файл, найдя его в проводнике.

```
Class1.cs – Блокнот
Файл  Изменить  Просмотр

using test_subsystem;
namespace test_subsystem {
    public class Class1 : Class2 {

        private int atr1;
        private string atr2;

        public Class1(){

        }

        ~Class1(){

        }

        public void op1(){

        }

        public void op2(){

        }

        public override void op3(){

        }

        public override int op4(){
            return 0;
        }

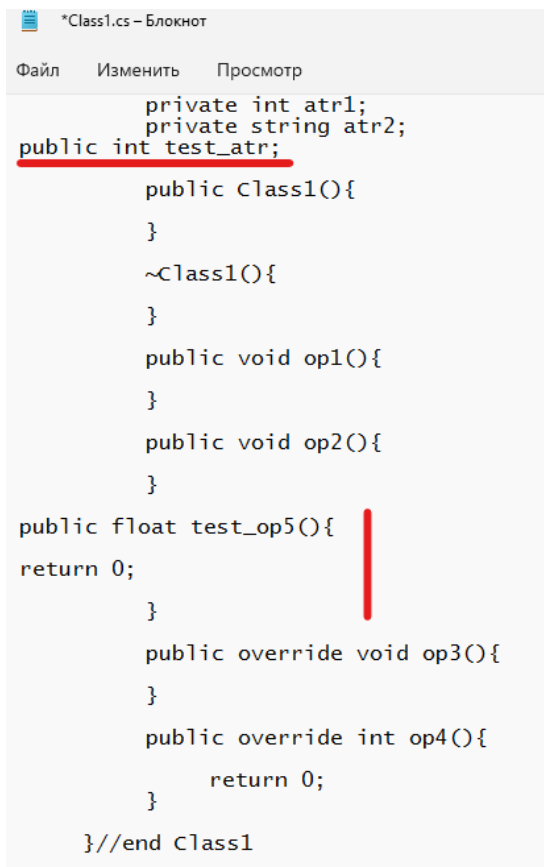
    }//end Class1
}

} //end namespace test_subsystem
```

Видно, что код был сгенерирован с соблюдением синтаксиса языка C#, а также с учетом всех указанных методов, связей наследования и т.д. Также были созданы конструкторы и деструкторы.

В качестве пространства имен используется название пакет, в котором расположены классы – test_subsystem.

Выполним обратную операцию. Запишем в коде новый метод и атрибут в класс Class1, а в Class2 наоборот удалим один из атрибутов.



```
*Class1.cs – Блокнот
Файл  Изменить  Просмотр

    private int atr1;
    private string atr2;
    public int test_atr;

    public Class1() {
    }

    ~Class1() {
    }

    public void op1() {
    }

    public void op2() {
    }

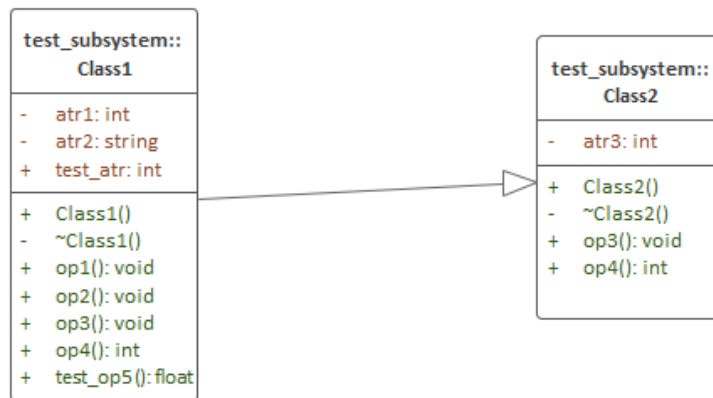
    public float test_op5() {
return 0;
    }

    public override void op3() {
    }

    public override int op4() {
        return 0;
    }

} //end class1
```

Теперь выполним **обратное проектирование** – для этого в меню выбрать Develop>Source Code>Synchronize> Synchronize Package... В открывшемся окне оставим настройки по умолчанию, затем согласиться с синхронизацией элементов. В итоге можно увидеть, что в классе Class1 появились те самые новые элементы с соблюдением типов данных и области доступности.



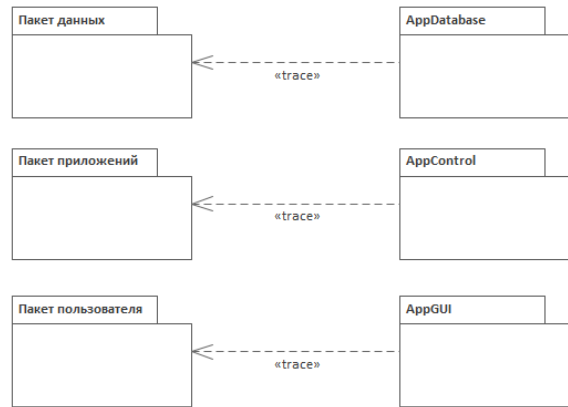
2.5. Трассировка пакетов в подсистемы

Подсистемы прикладного уровня определяют на основе пакетов анализа. Изначально пакеты анализа транслируются в подсистемы проектирования прикладных уровней, сервисные пакеты — в сервисные подсистемы.

Далее проводят уточнение и декомпозицию полученных подсистем по следующим правилам:

- часть пакета анализа, совместно используемую несколькими подсистемами, выделяют в отдельную подсистему;
- часть пакета анализа для многократного применения выделяют в отдельную подсистему, в том числе подсистему среднего уровня или уровня системного ПО;
- унаследованные подсистемы выделяют в отдельные подсистемы;
- выполняют декомпозицию подсистем для возможности разделения работ;
- осуществляют декомпозицию подсистем в целях разнесения их по узлам так, чтобы каждая подсистема была на отдельном узле.

В результате трассировки пакетов анализа в подсистемы была сформирована диаграмма трассировки.

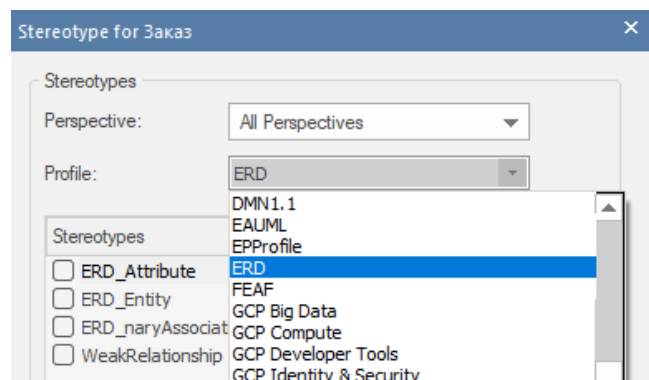


2.6. Создание инфологической модели

2.6.1. Создание вручную

Для создания инфологической модели вручную необходимо создать диаграмму классов и расположить на ней элементы-классы с соответствующими стереотипами:

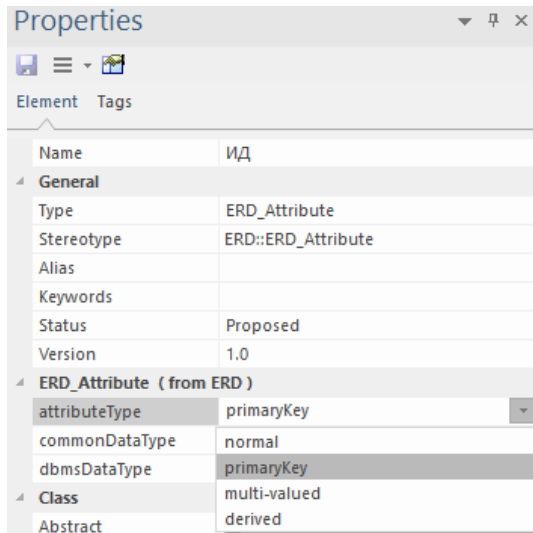
- Сущность – стереотип «ERD_Entity»;
- Атрибут – стереотип «ERD_Attribute»;
- Связь между атрибутом и сущностью – стереотип «ERD_Connector»;
- Связь между сущностями – стереотип «ERD_Relationship».



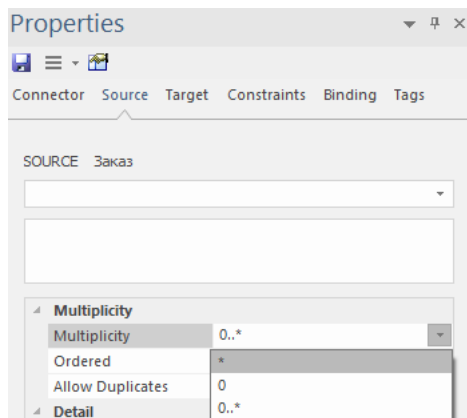
При применении стереотипов классы будут автоматически визуально изменяться в соответствии с нотацией ER-диаграммы.

Все элементы связываются связью Association из пункта Class Relationships в Toolbox.

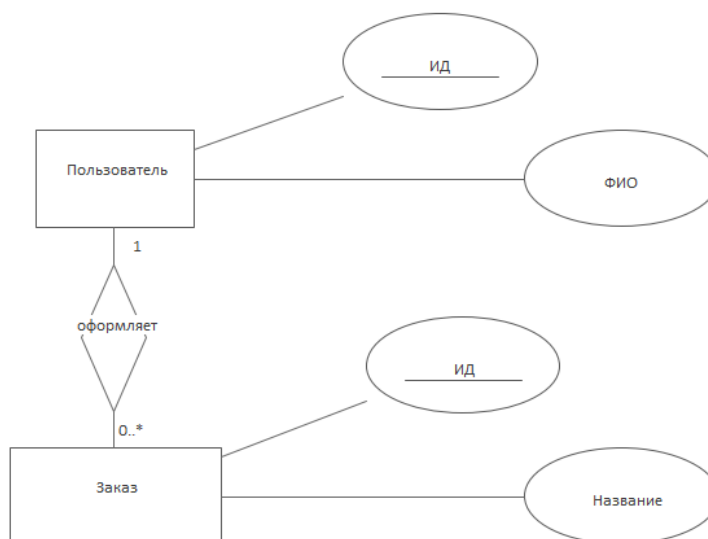
Для обозначения того, что атрибут является первичным ключом, необходимо в окне Properties установить соответствующее значение attributeType.



Для связей необходимо обозначить множественность, а также дать название каждой связи (внутри ромба на данной связи). Множественность также обозначается в окне Properties выбранной связи во вкладках Source и Target.

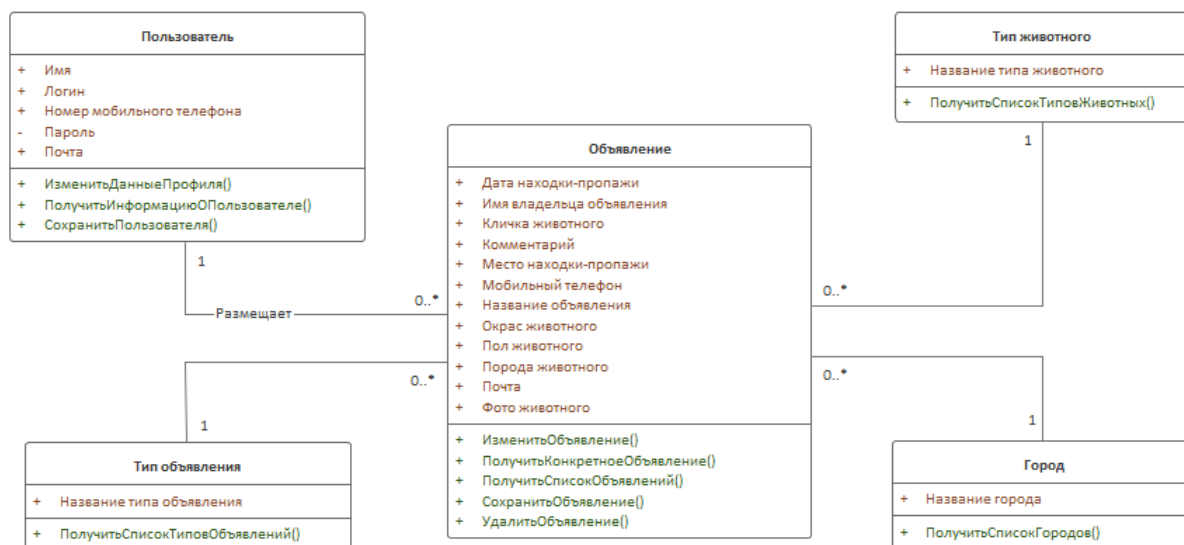


В итоге может быть получена простейшая инфологическая модель.



2.6.2. Генерация из диаграммы классов модели требований

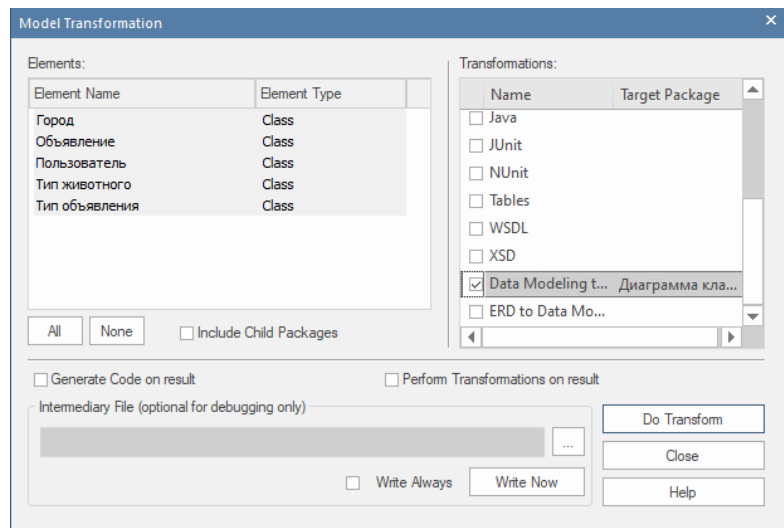
ER-диаграмму можно сгенерировать автоматически из уже созданной в процессе выполнения лабораторной работы №4 диаграммы классов. Ниже представлен пример такой диаграммы классов.



Также для генерации инфологической модели подойдет диаграмма классов сущностей.

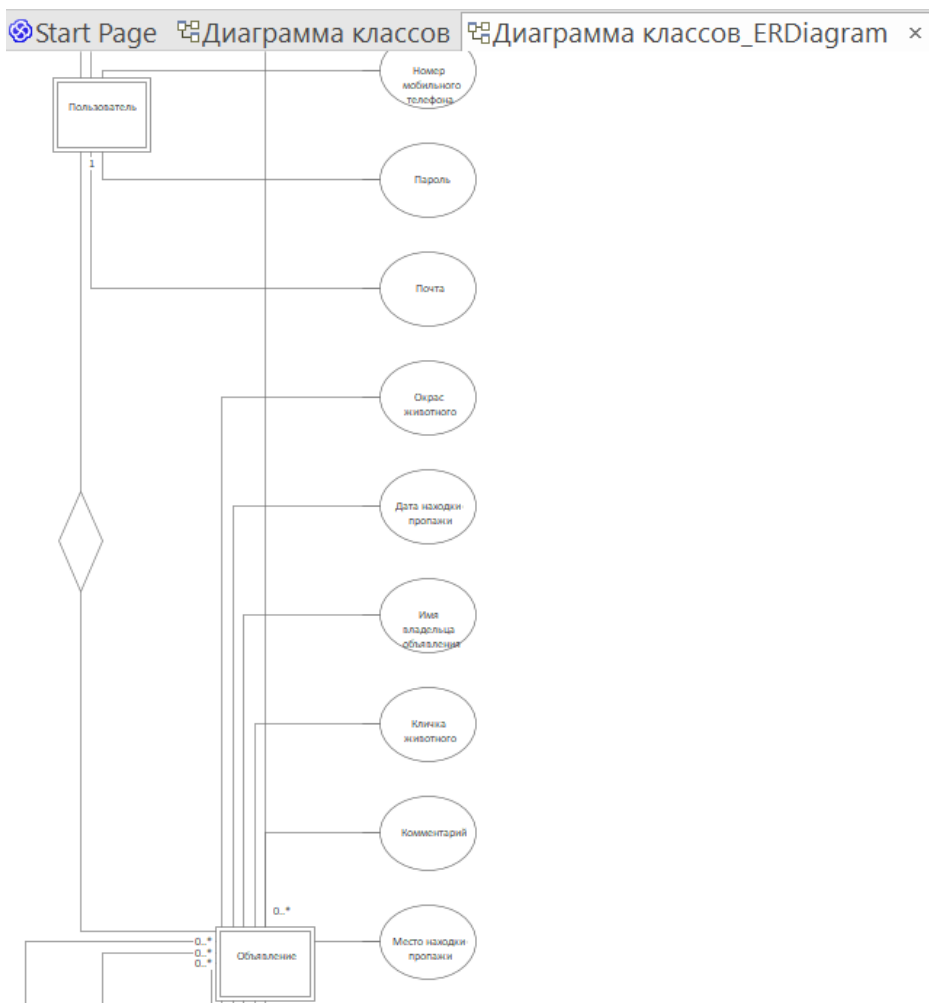
Для генерации ER-диаграммы из данной модели нужно выполнить следующие действия:

1. Выбрать в проводнике нужный package с диаграммой классов.
2. Перейти в пункт Design>Tools>Transform>Apply Transformation...
3. В открывшемся окне убедиться, что выбраны все классы диаграммы. Затем из списка возможных трансформаций выбрать Data Modeling to ERD. В открывшемся доп. окне выбрать целевой package, куда будет сохранена новая диаграмма и ее элементы.



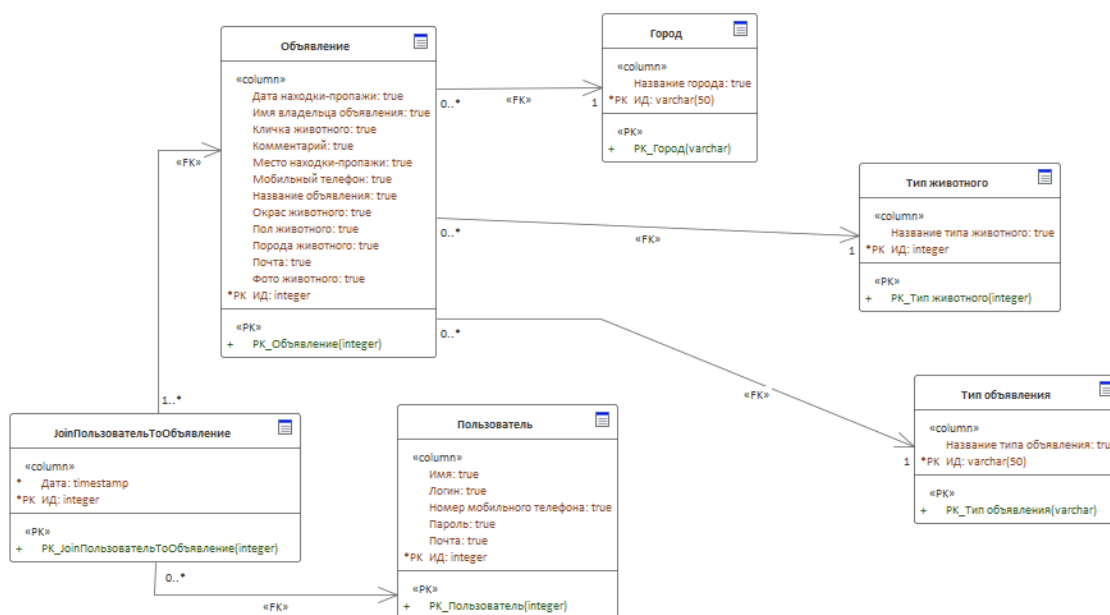
4. Нажать Do Transform.

В указанной целевой папке появится package, повторяющий название диаграммы и оканчивающийся на `_ERDiagram`. Обратите внимание, что, несмотря на наличие названия у связи Пользователь-Объявление на диаграмме классов, на инфологической модели связь не имеет названия (ромб пустой).



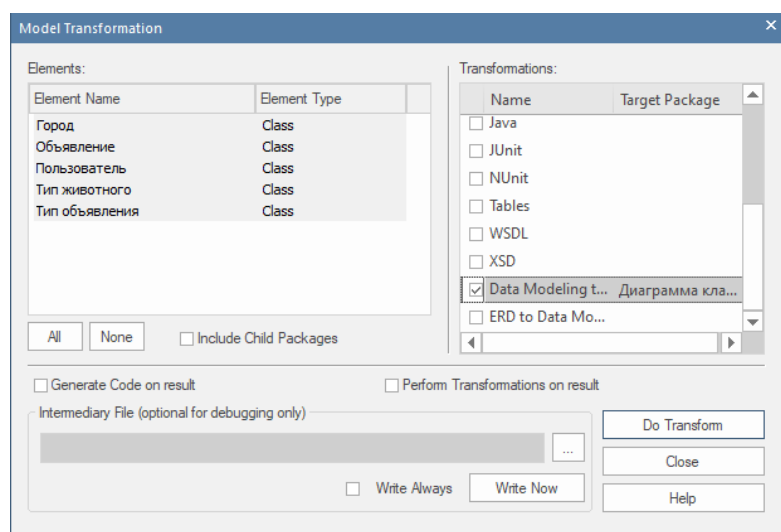
2.6.3. Генерация из даталогической модели

Пусть имеется следующая даталогическая модель:



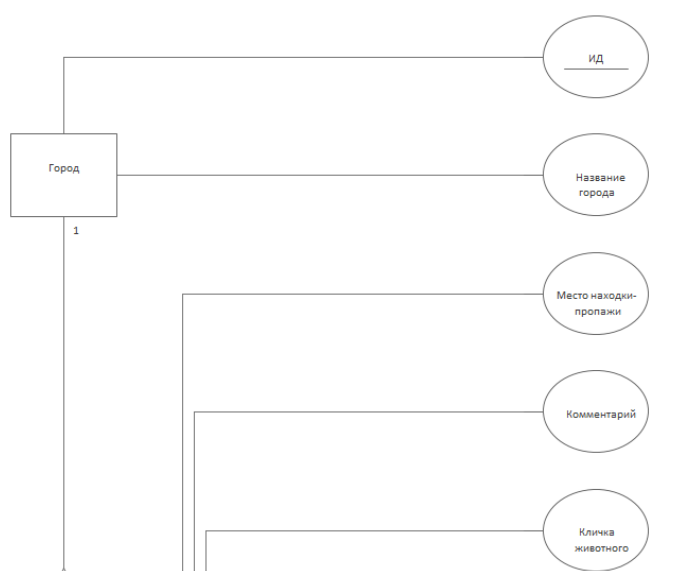
Генерация инфологической модели выполняется аналогичным образом, описанным в пункте выше:

1. Выбрать в проводнике нужный package с диаграммой классов.
2. Перейти в пункт Design>Tools>Transform>Apply Transformation...
3. В открывшемся окне убедиться, что выбраны все классы диаграммы. Затем из списка возможных трансформаций выбрать Data Modeling to ERD. В открывшемся доп. окне выбрать целевой package, куда будет сохранена новая диаграмма и ее элементы.



4. Нажать Do Transform.

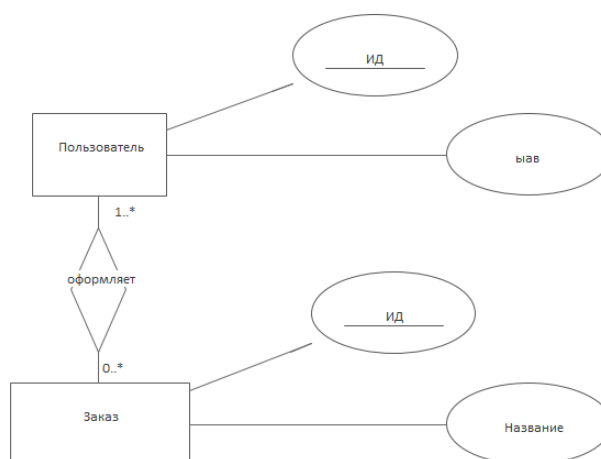
В указанной целевой папке появится package, повторяющий название диаграммы и оканчивающийся на _ERDiagram. Обратите внимание, что атрибуты, являющиеся первичными ключами в даталогической модели, будут отображаться подчеркнутыми в инфологической модели в соответствии с нотацией данной диаграммы.



2.7. Создание даталогической модели

2.7.1. Генерация из инфологической модели

Инфологическая модель была предварительно.

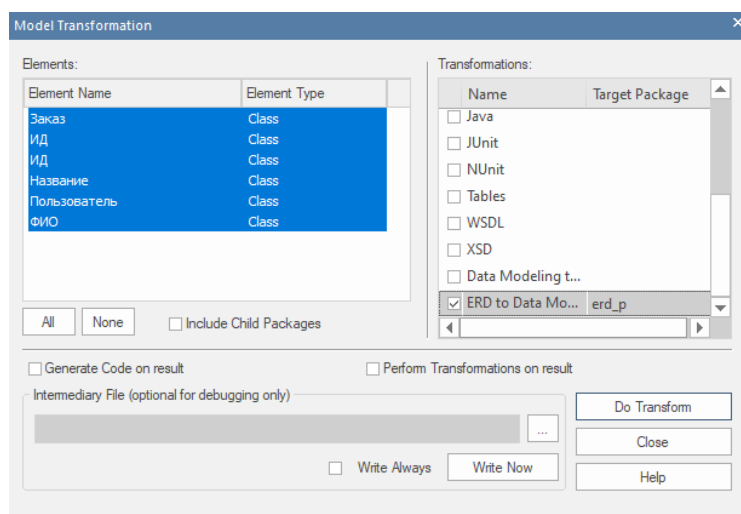


Обратите внимание на связь М-М между Пользователем и Заказом. В данном пункте работы это изменение было внесено специально. Далее будет продемонстрировано, что на даталогической модели такая связь будет

трансформирована в связующую таблицу для разрешения проблемы невозможности связи М-М в даталогической модели.

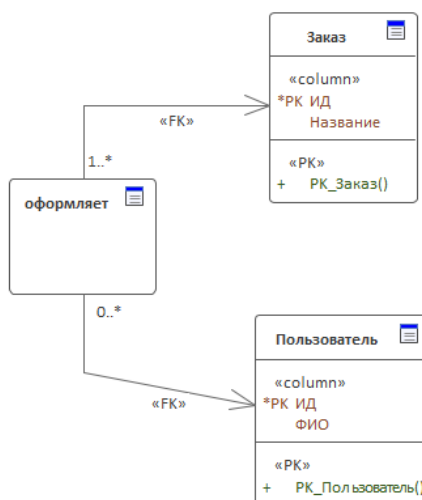
Для генерации требуется выполнить следующие действия:

1. Выбрать в проводнике нужный package с инфологической моделью.
2. Перейти в пункт Design>Tools>Transform>Apply Transformation...
3. В открывшемся окне убедиться, что выбраны все классы диаграммы. Затем из списка возможных трансформаций выбрать ERD to Data Modeling. В открывшемся доп. окне выбрать целевой package, куда будет сохранена новая диаграмма и ее элементы.

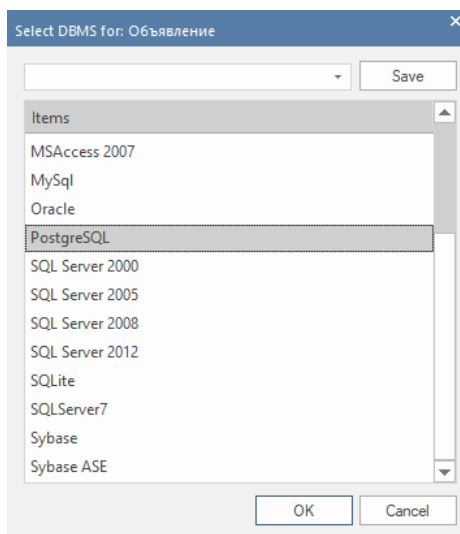


4. Нажать Do Transform.

В указанной целевой папке появится package, повторяющий название диаграммы и оканчивающийся на DataModeling. Обратите внимание, что связь М-М между Пользователем и Объявлением превратилась в отдельную таблицу с названием связи «Оформляет». Если бы название связи не было указано, то таблица называлась бы JoinПользовательТоЗаказ.

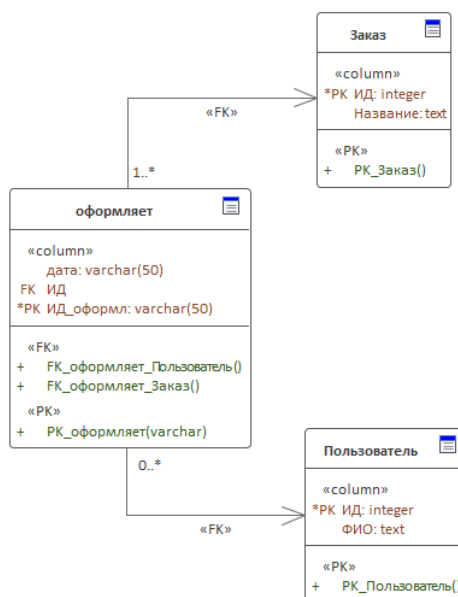


Для изменения типов данных в таблицах необходимо выбрать СУБД, это делается по клику по любой таблице.



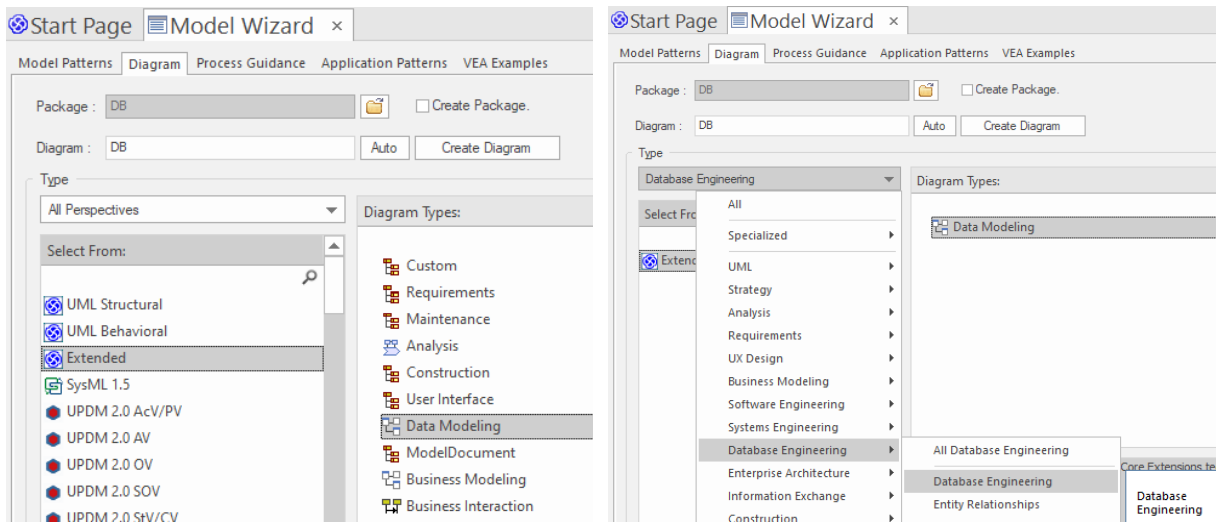
Для дальнейшей работы с данной моделью необходимо добавить хотя бы один атрибут в таблицу, созданную для разрешения конфликта связи М-М «Оформляет». Добавим поле «Дата». Автоматически внесены ограничения Foreign Key.

Также необходимо обозначить типы данных и дополнительно добавить свой первичный ключ для таблицы «Оформляет».

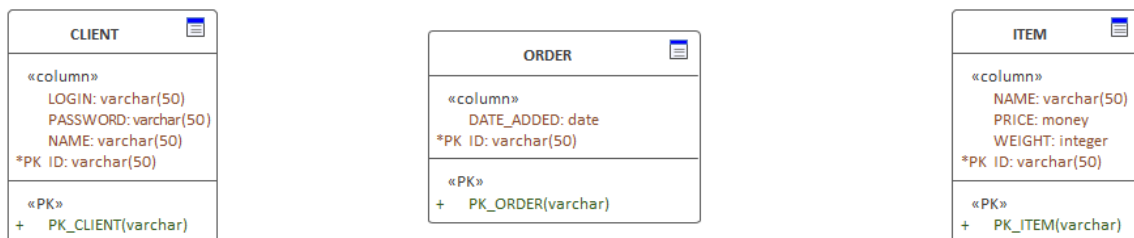


2.7.2. Создание вручную

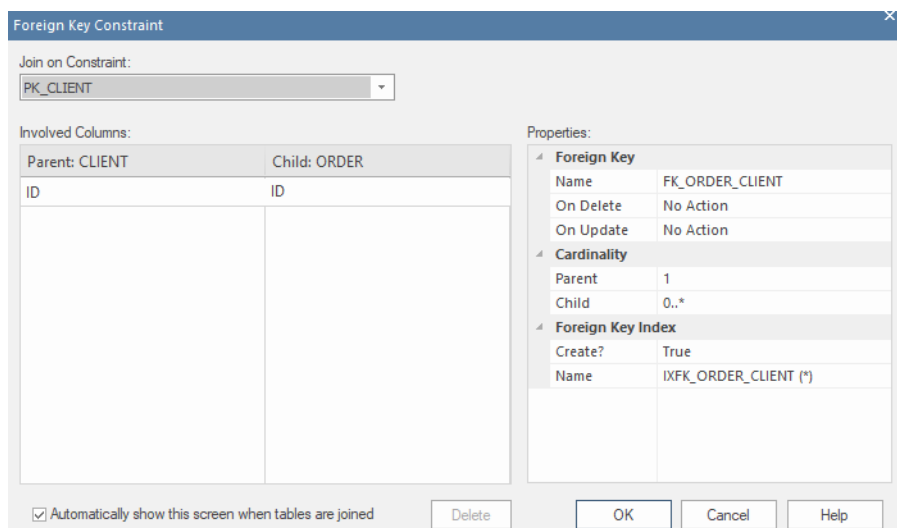
Создадим даталогическую модель (модель базы данных) вручную. Для этого в Model Wizard необходимо создать диаграмму Data Modeling из набора Extended. Также до данной диаграммы можно добраться, выбрав из выпадающего списка наборов Database Engineering.



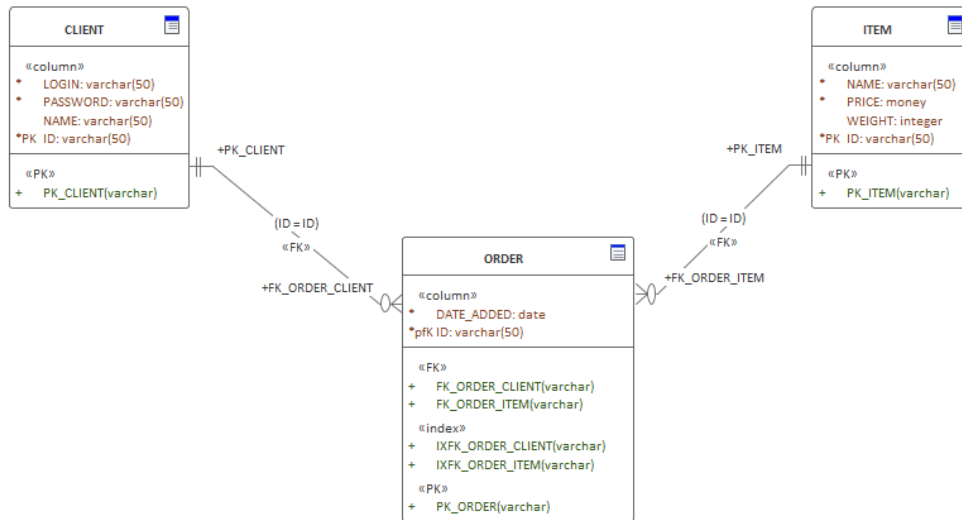
На диаграмме, используя элемент Table из Toolbox, постройте все необходимые таблицы и наполните их атрибутами, указав также первичный ключ и типы данных.



Теперь настроим связи между таблицами. Настроим связи-ассоциации, укажем множественность и ключи, по которым связываются таблицы:

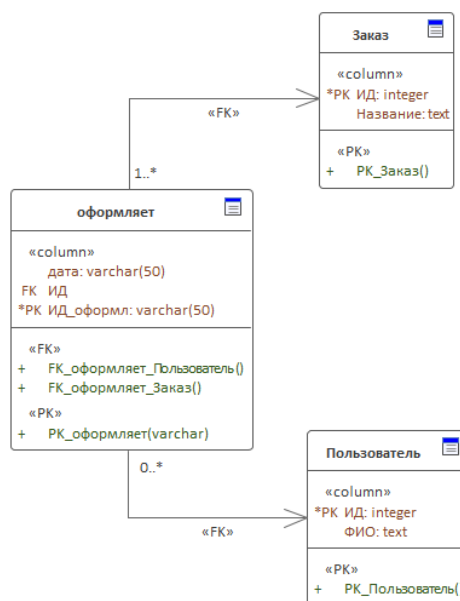


Таким образом, мы получили модель базы данных с указанием множественности, атрибутов таблиц и их ограничений.



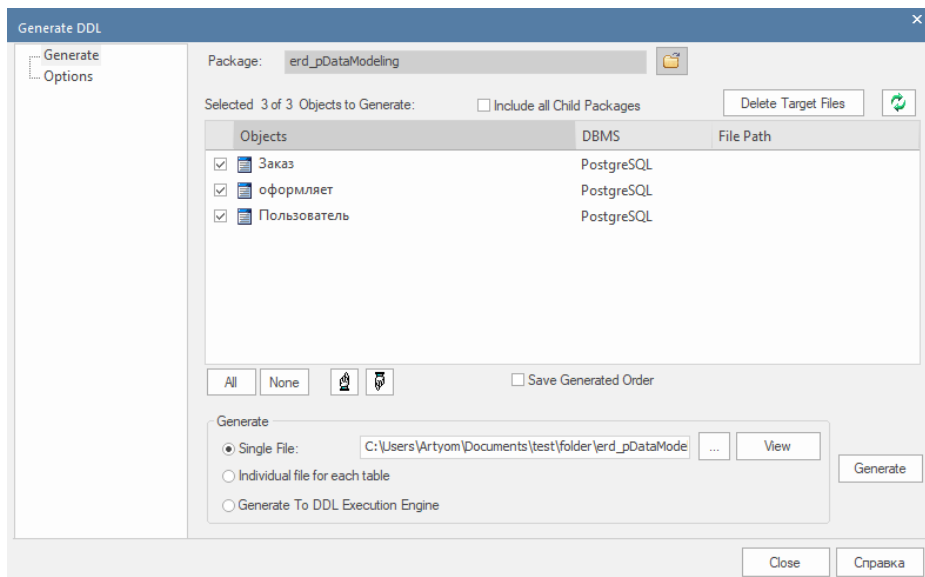
2.8. Генерация DDL-сценария

Пусть имеется следующая даталогическая модель (модель базы данных):



Для генерации DDL-сценария необходимо выполнить следующие действия:

1. Выбрав в обозревателе нужную диаграмму, перейти в Develop>Data Modeling>Generate.
2. В появившемся окне указать путь до будущего файла DDL-сценария.



3. Нажать Generate.

В результате получен файл с PostgreSQL-скриптом создания указанных таблиц и ограничений к ним.

```

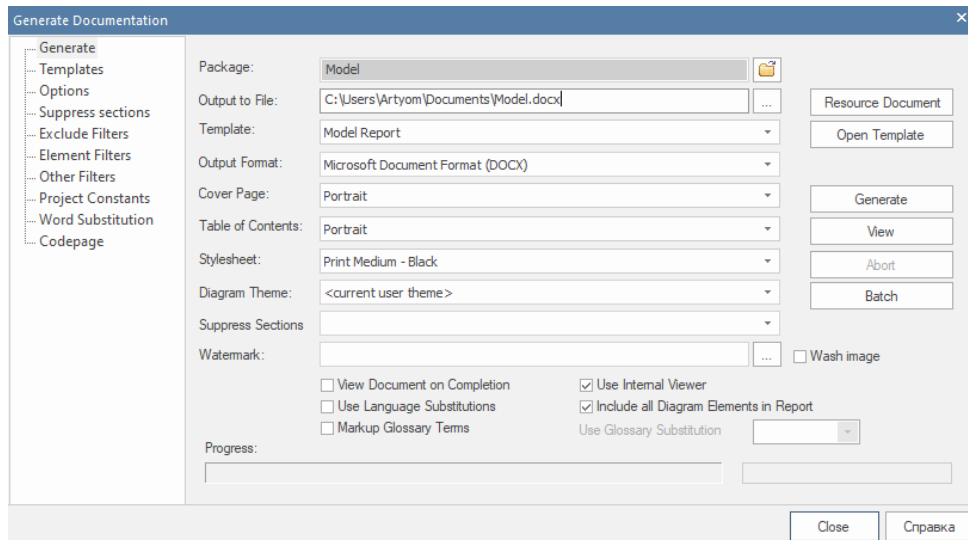
CREATE TABLE "Пользователь"
(
    "ИД" integer NOT NULL,
    "ФИО" text NULL
);
/* Create Primary Keys, Indexes, Uniques, Checks */
ALTER TABLE "Заказ" ADD CONSTRAINT "PK_Заказ"
PRIMARY KEY ("ИД")
;
ALTER TABLE "оформляет" ADD CONSTRAINT "PK_оформляет"
PRIMARY KEY ("ИД_оформл")
;
ALTER TABLE "Пользователь" ADD CONSTRAINT "PK_Пользователь"
PRIMARY KEY ("ИД")
;
/* Create Foreign Key Constraints */
ALTER TABLE "оформляет" ADD CONSTRAINT "FK_оформляет_Пользователь"
FOREIGN KEY ("ИД") REFERENCES "Пользователь" ("ИД") ON DELETE No Action ON UPDATE No Action
;
ALTER TABLE "оформляет" ADD CONSTRAINT "FK_оформляет_Заказ"
FOREIGN KEY ("ИД") REFERENCES "Заказ" ("ИД") ON DELETE No Action ON UPDATE No Action
;

```

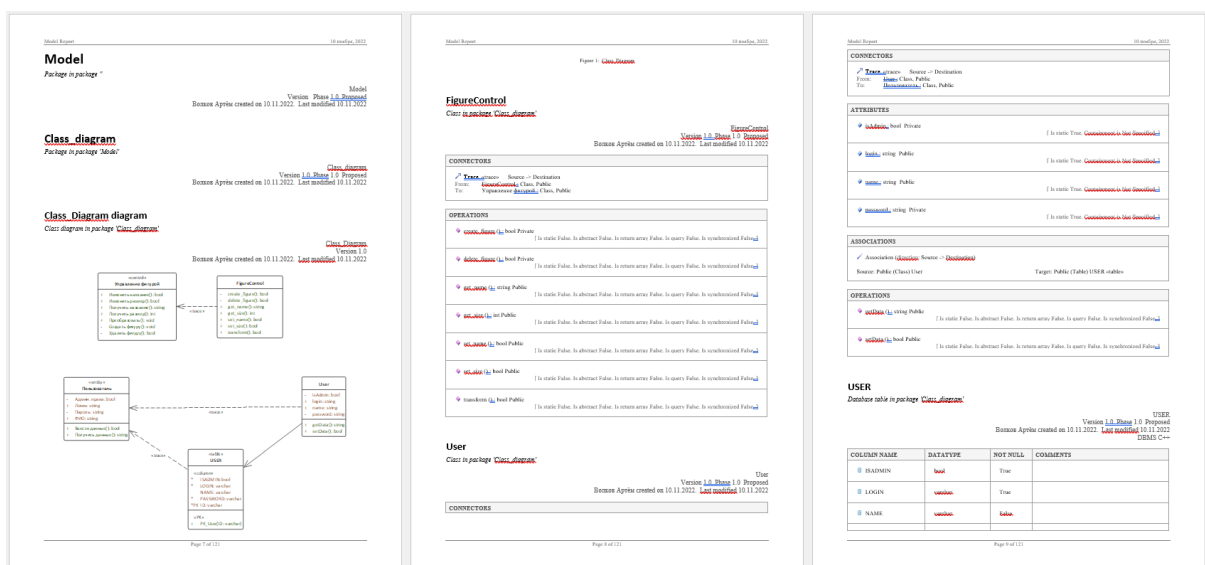
2.9. Генерация документации

Enterprise Architect располагает широкими возможностями по документированию проекта и автоматическому созданию документации. Для того, чтобы сгенерировать документацию нужно перейти в пункт Publish > Model Reports > Report Builder > Generate Documentation > Generate.

Введите необходимую информацию в открывшемся окне, нажмите кнопку Generate и дождитесь окончания. Обратите внимание, что документацию можно генерировать не только для всего проекта, но и для отдельных пакетов (package).



Результирующую документацию можно открыть в проводнике и изучить.



В Приложении 1 представлено описание всех пунктов настройки генерации документации.

2.10. Проверка целостности данных проекта

Репозиторий (проект), созданный в Enterprise Architect, хранится как реляционная база данных [3]. Независимо от используемого ядра базы данных (MS Jet для EAP, MS SQL, MySQL, Oracle или других файлов) для хранения моделей используется одна и та же схема базы данных. Данные реляционной базы данных должны быть непротиворечивыми.

Эта непротиворечивость может быть обеспечена техническим способом путем введения определенных ограничений, после чего механизм базы данных обеспечивает соответствие определенным правилам.

Если репозиторий содержит данные, которые не согласуются с программой EA, EA не информирует нас сообщениями об ошибках. EA просто представляет данные как можно лучше. Вероятно, в коде есть механизмы для отображения или даже изменения таких данных.

Несоответствия в репозитории могут отрицательно сказаться на производительности, что может ощущаться в более сложных операциях, таких как, например, сравнение моделей.

Данную функцию проверки целостности следует использовать в основном при наличии общего репозитория, а также в случае локального проекта, разрабатываемого одним человеком. Возможно, только с той разницей, что вероятность несоответствия выше в первом случае.

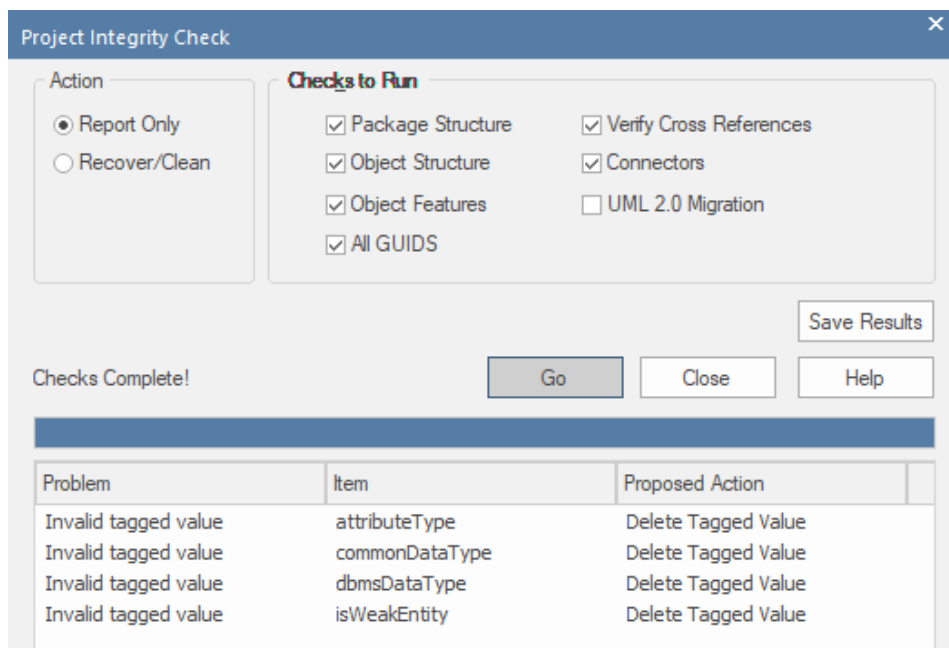
Если у вас произошел сбой импорта XMI, сбой сети или другое непредвиденное событие, нарушающее целостность информации в модели, рекомендуется запустить функцию проверки целостности проекта, чтобы убедиться, что данные вашего проекта структурно завершены.

Вы можете выбрать различные элементы для проверки. При проверке целостности проверяются все записи базы данных и гарантируется отсутствие «осиротевших» записей, неточных или неустановленных идентификаторов. **Эта функция НЕ проверяет соответствие UML, а только отношения данных и структуру репозитория.**

Вы можете сначала запустить средство проверки целостности в режиме отчета, чтобы узнать, нужно ли что-то исправить, а затем снова запустить его в режиме «восстановить/очистить».

Когда Enterprise Architect «очищает» модель, он пытается восстановить потерянные пакеты и элементы и создает новый пакет на корневом уровне модели с именем `_recovered_`. Проверьте все найденные элементы и, при необходимости, перетащите их обратно на место. Если они не нужны, удалите их.

Выполним проверку через `Configure > Model > Integrity > Project Integrity`.

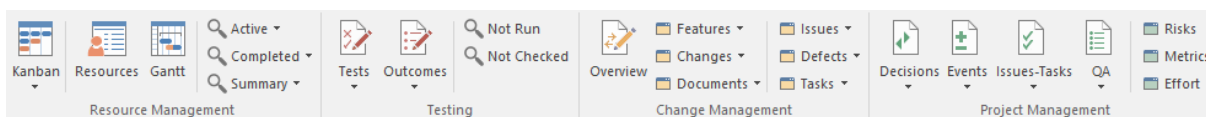


По нажатию Save Results можно сохранить результаты проверки в виде текстового .log-файла.

2.11. Статистика проекта

Enterprise Architect располагает широким функционалом управления проектом.

На ленте «Construct» представлены инструменты и средства для визуализации и управления задачами и ресурсами, необходимыми для построения системы, от распределения человеческих ресурсов по задачам до мощных досок Канбан и диаграммы Ганта для визуализации хода выполнения работы. Команды могут сотрудничать непосредственно в инструменте и управлять широким спектром вариантов элементов модели, включая проблемы и дефекты, а также определять тесты и фиксировать результаты.



Для просмотра статистики по содержимому проекта необходимо перейти в пункт: Construct > Project Management > QA

Нажмите на этот пункт, чтобы отобразить два варианта; во-первых, отобразить представление QA Report («Отчеты о контроле качества»), в котором есть отдельные вкладки, позволяющие запускать и отображать следующие системные отчеты о состоянии:

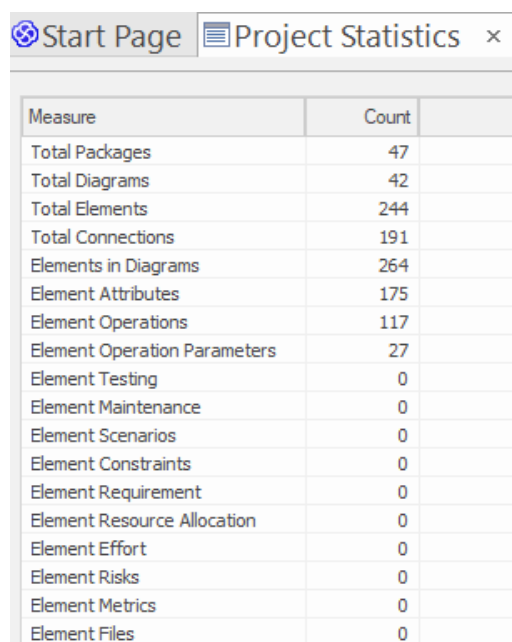
- Детали тестирования
- Сведения о техническом обслуживании

- Сведения о зависимости
- Детали реализации
- Метрики вариантов использования

Первый вариант предполагает широко документированный проект. Ограничимся в рамках лабораторной работы вторым вариантом.

Второй вариант - отобразить лист «Статистика проекта», в котором перечислены номера пакетов, диаграмм, элементов, соединителей, элементов каждого типа и характеристик элементов каждого типа в проекте. Для определенных элементов, таких как риски элемента, метрики и трудозатраты, если нет экземпляров этого конкретного типа объекта, он все равно отображается в списке, но с нулевым счетчиком.

Отобразим статистику проекта:



| Measure | Count |
|------------------------------|-------|
| Total Packages | 47 |
| Total Diagrams | 42 |
| Total Elements | 244 |
| Total Connections | 191 |
| Elements in Diagrams | 264 |
| Element Attributes | 175 |
| Element Operations | 117 |
| Element Operation Parameters | 27 |
| Element Testing | 0 |
| Element Maintenance | 0 |
| Element Scenarios | 0 |
| Element Constraints | 0 |
| Element Requirement | 0 |
| Element Resource Allocation | 0 |
| Element Effort | 0 |
| Element Risks | 0 |
| Element Metrics | 0 |
| Element Files | 0 |

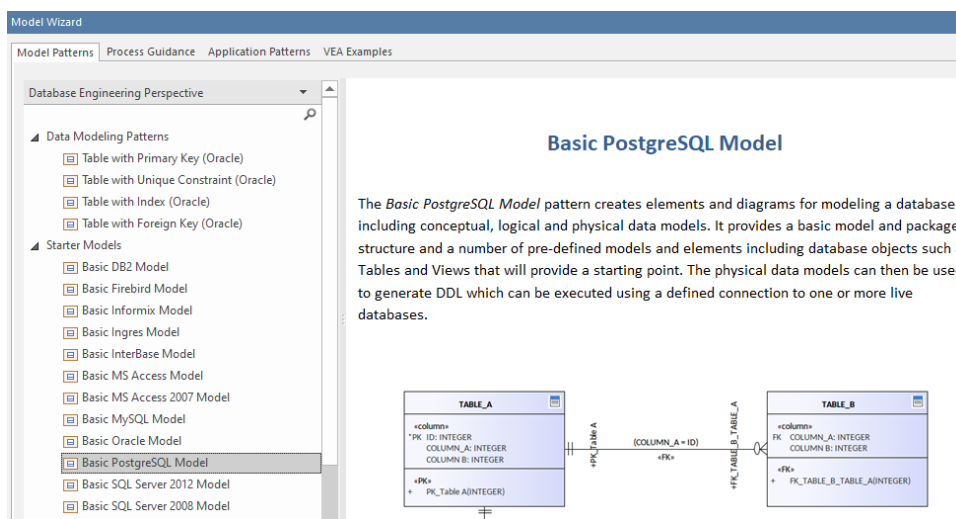
2.12. Пример проектирования базы данных

Данный раздел посвящен ознакомлению с расширенными возможностями проектирования базы данных в среде Enterprise Architect.

2.12.1. Шаблон базы данных

Database Builder — это адаптированный интерфейс для средства моделирования данных; все задачи моделирования, связанные с базой данных, могут выполняться в одном месте. Интерфейс и связанные с ним экраны содержат только информацию, относящуюся к моделированию данных, что оптимизирует и упрощает процесс моделирования.

Для проектирования базы данных перейдите в Develop>Data Modeling>Database Builder. Создайте новую базу данных. В открывшемся окне выберите шаблон Basic PostgreSQL Model.

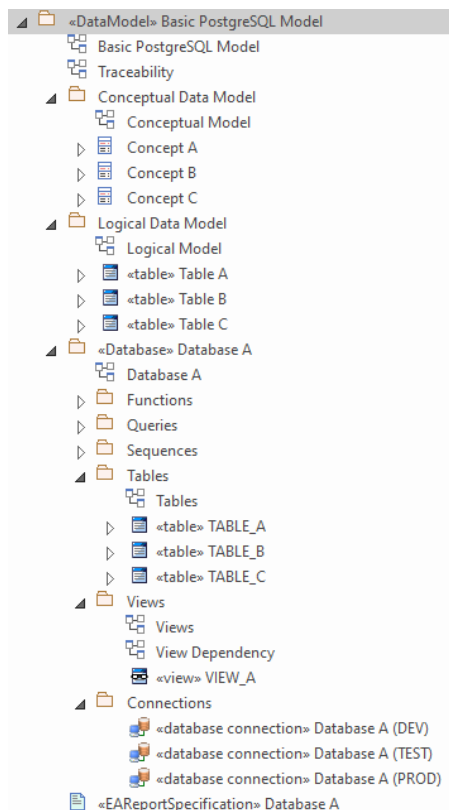


Шаблон Basic PostgreSQL Model создает элементы и диаграммы для моделирования базы данных, включая концептуальные, логические и физические модели данных. Он предоставляет базовую модель и структуру пакетов, а также ряд predefined моделей и элементов, включая объекты базы данных, такие как таблицы и представления, которые обеспечат отправную точку. Физические модели данных затем могут быть использованы для создания DDL, который может быть выполнен с использованием определенного соединения с одной или несколькими действующими базами данных.

Получаемый проект базы данных содержит исчерпывающий набор информации включающий в себя:

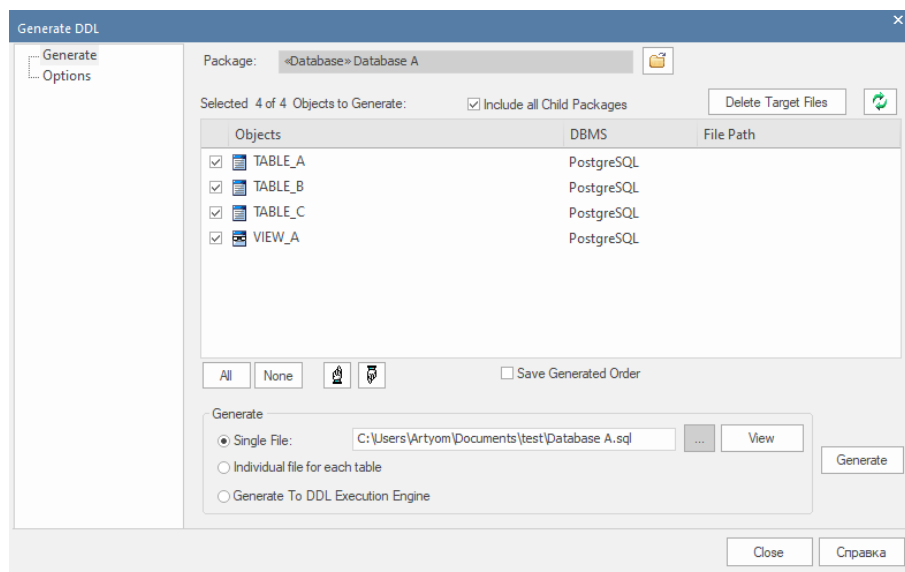
- Basic PostgreSQL Model – обзор основных разделов проекта базы данных;
- Traceability – порядок «эволюции» проектирования базы данных: таблица с инфологической модели (Concept A), с даталогической (Table A) и уже с диаграммы базы данных (TABLE_A);
- Conceptual Data Model – инфологическая модель;
- Logical Data Model – даталогическая модель;
- «Database» Database A – подробный проект базы данных, включающий в себя:
 - Functions – функции (в соответствии с PostgreSQL);
 - Queries – запросы;
 - Sequences;
 - Tables – таблицы и их диаграмму (по сути, даталогическую модель или диаграмму базы данных);

- Views – представления;
- Connections – связи с серверами баз данных;
- «EARReportSpecification» - шаблон генерации документации.



2.12.2. Генерация DDL-сценария

Аналогичным образом сгенерируем DDL-сценарий, находясь в обозревателе на уровне папки «DataModel» Basic PostgreSQL Model.

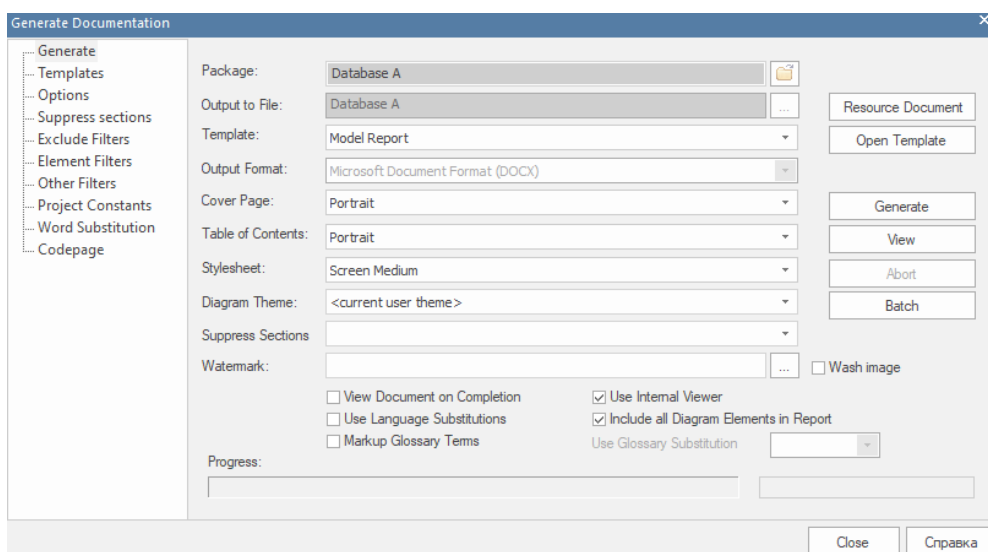
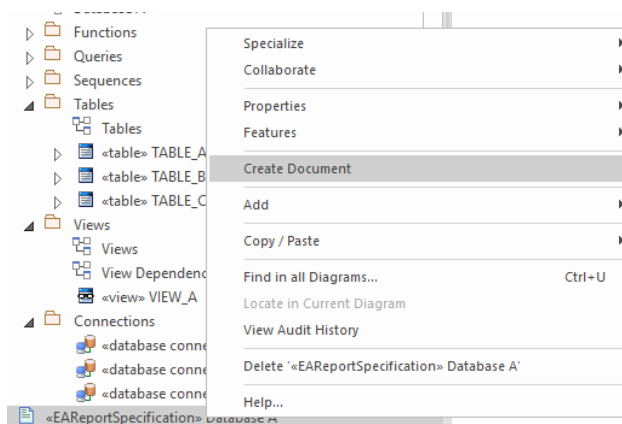


Обратите внимание, что, несмотря на нахождение указателя на более высоком уровне в обозревателе, среда автоматически находит пакет со стереотипом «database» и загружаем из него определенные элементы со стереотипами «table» и «view».

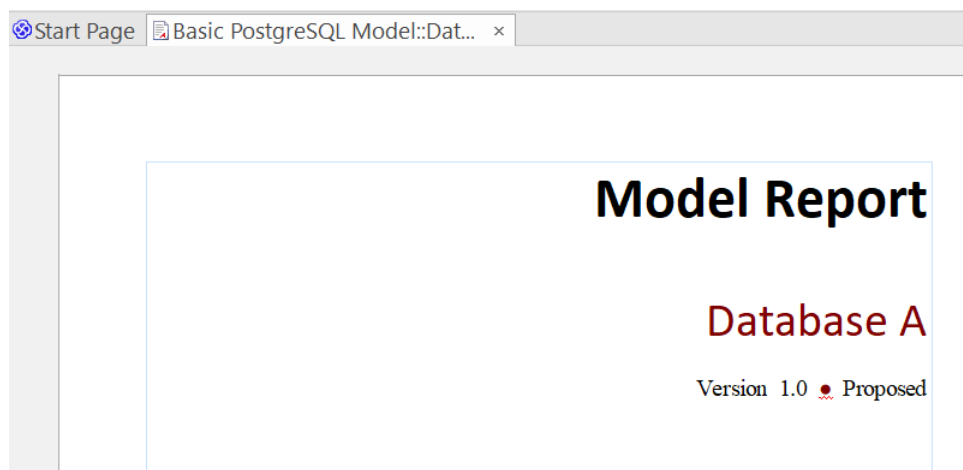
```
Database A.sql – Блокнот
Файл  Изменить  Просмотр
)
;
CREATE TABLE "TABLE_C"
(
  "COLUMN_A" integer NULL,
  "COLUMN_B" integer NULL
)
;
/* Create Primary Keys, Indexes, Uniques, Checks */
ALTER TABLE "TABLE_A" ADD CONSTRAINT "PK_Table A"
PRIMARY KEY ("ID")
;
/* Create Foreign Key Constraints */
ALTER TABLE "TABLE_B" ADD CONSTRAINT "FK_TABLE_B_TABLE_A"
FOREIGN KEY ("COLUMN_A") REFERENCES "TABLE_A" ("ID") ON DELETE No Action ON UPDATE No Action
;
ALTER TABLE "TABLE_C" ADD CONSTRAINT "FK_TABLE_C_TABLE_A"
FOREIGN KEY ("COLUMN_A") REFERENCES "TABLE_A" ("ID") ON DELETE No Action ON UPDATE No Action
;
/* Create Views */
CREATE VIEW "VIEW_A" AS
/* SQL statements defining VIEW_A */
;
```

2.12.3. Генерация документации

Создать документацию можно, кликнув ПКМ по шаблону «EAReportSpecification».



По нажатию кнопки View можно посмотреть материал сразу внутри Enterprise Architect.



3. КОНТРОЛЬНЫЕ ВОПРОСЫ


1. Для чего выполняется трассировка классов, подсистем?
2. Как выполняют проектирование класса? Что будет его результатом?
3. Как определяют и описывают атрибуты, операции, методы и связи классов?
4. Что такое прямое и обратное проектирование? Как оно выполняется?
5. Что такое инфологическая модель? Как ее создать?
6. Что такое даталогическая модель? Как ее создать?
7. Что такое DDL-сценарий? Как его сгенерировать?
8. Каким образом пакеты и класса анализа транслируются в подсистемы и классы проектирования? На что влияет стереотип класса анализа?
9. В чем цель проектирования подсистем?
10. Что такое документация проекта? Как создать ее?
11. Что такое проверка целостности проекта? Как ее автоматически выполнить?
12. Что содержит статистика проекта? Как ее посмотреть?
13. Для чего выполняется проектирование базы данных? Что содержит проект базы данных?


4. СПИСОК ИСТОЧНИКОВ

1. Sparx Systems – Текст. Изображение.: электронные // Sparx Systems : [сайт]. – URL: <https://sparxsystems.com/> (дата обращения 15.06.2022)
2. Фаулер, Мартин. Архитектура корпоративных программных приложений.: Пер. с англ. — М.: Издательский дом "Вильямс", 2006. — 544 с.: ил. — Парал. тит. англ. ISBN 5-8459-0579-6 (рус.)
3. Checking the integrity of an EA repository – Текст. Изображение.: электронные //Expert IT : [сайт]. – URL: <https://www.expert-it.com/en/checking-integrity-ea-repository> (дата обращения 15.06.2022)

5. ПРИЛОЖЕНИЕ 1

Описание пунктов генерации документации

| Пункт меню | Описание |
|---|--|
| Model Document Root Element Root Package | <p>Проверьте имя элемента, выбранного в окне браузера, диаграмме, списке диаграмм, диспетчере спецификаций или поиске модели.</p> <p>Если это специально созданный элемент Model Document для виртуального документа, полем будет 'Model Document'.</p> <p>В противном случае в этом поле указывается выбранный элемент иерархии, по которому будет сообщено; то есть корневой элемент или корневой пакет.</p> |
| Output to File | <p>Введите или выберите расположение и имя файла для созданной документации.</p> <p>Нажмите на значок  чтобы открыть проводник.</p> |
| Template | <p>Щелкните стрелку раскрывающегося списка и выберите имя шаблона, который будет использоваться для отчета о документе.</p> <p>Вы можете выбрать любой шаблон из выпадающего списка, который разделен на пользовательские (или пользовательские) шаблоны и системные шаблоны. Если вы загрузили технологию ЦРТ, содержащую шаблоны документов, они также перечислены под названием технологии.</p> <p>Вы также можете сразу же создать новый шаблон, выбрав опцию «<новый шаблон>».</p> <p>Система проверяет все папки шаблонов для указанного шаблона в порядке Пользовательские шаблоны > Шаблоны технологий > Системные шаблоны. Если под одним и тем же именем доступны разные шаблоны, система использует первый найденный в этой последовательности.</p> |
| Output Format | <p>Нажмите на стрелку раскрывающегося списка и выберите формат вывода отчета. По умолчанию используется формат RTF (. RTF), но вы также можете выбрать:</p> <ul style="list-style-type: none"> • Формат переносимого документа (.PDF) или • Формат документов Майкрософт (.DOCX) |
| Cover Page | <p>Щелкните стрелку раскрывающегося списка и выберите название определения титульной страницы, которое будет применено к отчету по документу.</p> <p>В раскрывающемся списке можно выбрать любое определение, будь то стандартный формат или страница, разработанная пользователем.</p> <p>Используйте значение по умолчанию <none> , чтобы не включать титульную страницу или использовать титульную страницу,</p> |

| Пункт меню | Описание |
|-------------------|---|
| | встроенную в выбранный пользовательский шаблон. |
| Table of Contents | <p>Щелкните стрелку раскрывающегося списка и выберите имя определения оглавления, которое будет применено к отчету о документе.</p> <p>В раскрывающемся списке можно выбрать любое определение, будь то стандартный формат оглавления или формат, разработанный пользователем.</p> <p>Используйте значение по умолчанию <none> , чтобы не включать оглавление или использовать оглавление, встроенное в выбранный пользовательский шаблон.</p> |
| Stylesheet | <p>Щелкните стрелку раскрывающегося списка и выберите имя таблицы стилей, которая будет применена к отчету о документе.</p> <p>Из раскрывающегося списка можно выбрать любую таблицу стилей, будь то стандартное определение или набор пользовательских стилей.</p> <p>Используйте значение по умолчанию '<none>', чтобы не использовать таблицу стилей, а вместо этого применить стили, определенные в шаблоне, выбранном в поле 'Шаблон'.</p> |
| Diagram Theme | <p>Щелкните стрелку раскрывающегося списка и выберите тему диаграммы, чтобы применить цвет, толщину линии и стиль шрифта к диаграммам элементов в отчете, если эти элементы используют значения по умолчанию.</p> <p>Если элемент имеет индивидуальный внешний вид, назначенный с помощью, например, диалогового окна «Внешний вид по умолчанию», этот внешний вид переопределяет параметры темы.</p> <p>Параметром по умолчанию является тема диаграммы, которую вы в настоящее время установили в своих «Локальных» параметрах. Однако вы можете выбрать любую другую тему только для этого отчета. После создания отчета параметр поля возвращается к теме по умолчанию.</p> |
| Suppress Sections | <p>Нажмите на стрелку раскрывающегося списка и выберите из списка Профилей подавления, созданных вами или другими пользователями модели, на вкладке «Подавить раздел» диалогового окна «Создать документацию».</p> <p>Каждый профиль подавления определяет разделы отчета, которые могут быть опущены при создании текущего отчета. Это поможет вам адаптировать отчет для конкретной аудитории по генерации.</p> |
| Watermark | <p>Введите местоположение изображения, которое будет использоваться в качестве водяного знака в документе, или нажмите на значок  для открытия проводника.</p> |

| Пункт меню | Описание |
|--|--|
| Wash image | Если выбрано изображение водяного знака, установите этот флажок, чтобы применить стирку к изображению перед его применением к документу. Это делает изображение светлее, чтобы оно не отвлекало от содержимого отчета, напечатанного поверх него. |
| View Document On Completion | Установите этот флажок, чтобы открыть отчет о документе сразу после его создания. |
| Use Language Substitutions | <p>Установите этот флажок, чтобы включить подстановку слов на пользовательском языке.</p> <p>Снимите флажок, чтобы отключить подстановки слов на пользовательском языке.</p> |
| Markup Glossary Terms | <p>Установите этот флажок, чтобы применить к созданному отчету RTF все стили, примененные к любым терминам глоссария через текст Notes элемента.</p> <p>Стиль применяется к термину глоссария в соответствии с его типом термина глоссария.</p> |
| Substitution of Glossary Terms | <p>(Доступно, если выбран параметр Термины глоссария разметки.)</p> <p>Щелкните стрелку раскрывающегося списка и выберите тип термина. Для всех терминов глоссария такого типа, где бы термин ни встречался в докладе, он будет заменен его значением.</p> <p>Например, если вы выберете тип термина «издание», в отчете термины «Corp», «Uni» и «Ult» будут заменены значениями «Corporate», «Unified» и «Ultimate» соответственно.</p> |
| Use Internal Viewer | <p>Установите этот флажок, чтобы включить кнопку Вид для запуска созданного документа во внутреннем средстве просмотра Enterprise Architect.</p> <p>Снимите этот флажок, чтобы включить кнопку Вид для запуска созданного документа в средстве просмотра файлов MS Windows по умолчанию.</p> |
| Include all Diagram Elements in Report | <p>Установите этот флажок, чтобы включить в отчет элементы из внешних пакетов, на которые ссылается диаграмма, для каждой диаграммы, по которой выполняется поиск в отчете; флажок по умолчанию установлен.</p> <p>Чтобы включить внешние элементы в документ, установите также флажки 'Package.Diagram.Element' и 'Package.Element' в текущем шаблоне.</p> <p>Если внешние элементы должны быть включены только для определенных схем, снимите этот флажок и - только для схем, для которых вы хотите включить внешние элементы в отчет - для каждой схемы установите флажок "Документировать каждый содержащийся</p> |

| Пункт меню | Описание |
|-------------------|---|
| | <p>элемент в RTF" в свойствах схемы.</p> <p>Если оба параметра не выбраны или если ни один из флажков «Package.Diagram.Element» или «Package.Element» не установлен в шаблоне, документируются только элементы в текущем пакете.</p> |
| Progress | Отображение хода создания документа с помощью сообщений о состоянии и все более заполняемой строки состояния. |
| Resource Document | Нажмите на эту кнопку, чтобы сохранить заданные параметры в виде определения документа. |
| Open Template | <p>Нажмите на эту кнопку, чтобы отредактировать выбранный шаблон с помощью редактора шаблонов документов.</p> <p>Вы можете редактировать только пользовательские шаблоны, а не стандартные шаблоны, поставляемые с системой. Если выбран системный шаблон, эта кнопка неактивна.</p> |
| Generate | Нажмите на эту кнопку, чтобы сгенерировать документ (после того, как вы установили все необходимые параметры, на всех вкладках диалогового окна). |
| View | <p>Нажмите на эту кнопку, чтобы запустить сгенерированный документ в:</p> <ul style="list-style-type: none"> • Средство просмотра файлов MS Windows по умолчанию, или • Внутреннее средство просмотра Enterprise Architect, если установлен флажок "Использовать внутреннее средство просмотра" |
| Abort | Нажмите на эту кнопку, чтобы отменить генерацию отчета; появится запрос на подтверждение отмены отчета. |