



Руководство пользователя



Подготовка и использование фотореалистичного окружения с помощью модуля UM Scene

Рассмотрены специализированные возможности программного комплекса "Универсальный механизм" по созданию и использованию фотореалистичных сцен

Оглавление

30. ПОДГОТОВКА И ИСПОЛЬЗОВАНИЕ ФОТОРЕАЛИСТИЧНОГО ОКРУЖЕНИЯ С ПОМОЩЬЮ МОДУЛЯ UM SCENE.....	1-3
30.1. ВВЕДЕНИЕ	1-3
30.2. НАЧИНАЕМ РАБОТАТЬ	1-4
30.3. СТРУКТУРА БАЗЫ 3D МОДЕЛЕЙ	1-5
30.4. ВЗАИМОДЕЙСТВИЕ СЦЕН С МОДЕЛЯМИ УМ.....	1-6
30.5. ПОЛОЖЕНИЕ И ОРИЕНТАЦИЯ ОБЪЕКТОВ СЦЕНЫ	1-7
30.6. ДЕСЯТИЧНЫЙ РАЗДЕЛИТЕЛЬ И РЕГИСТР ИМЕН УЗЛОВ И АТТРИБУТОВ	1-7
30.7. СОЗДАНИЕ СЦЕН.....	1-8
30.7.1. Формат файла сцены.....	1-8
30.7.2. Подложки.....	1-10
30.7.3. Скайбоксы.....	1-11
30.7.4. Создание и редактирование файлов сцен	1-12
30.8. РАБОТА С ПОЛЬЗОВАТЕЛЬСКОЙ БАЗОЙ 3D ОБЪЕКТОВ	1-14
30.8.1. Экспорт объекта из Autodesk 3ds Max	1-14
30.8.1.1. Объединение объектов	1-15
30.8.1.2. Экспорт объектов.....	1-18
30.8.1.3. Типичные проблемы экспорта и их исправление	1-21
30.8.2. Создание пользовательской базы объектов.....	1-24
30.8.2.1. Создание ZIP-архива	1-24
30.8.2.2. Создание XML-файла.....	1-25
30.8.2.3. Связь ZIP-архивов и XML-файлов.....	1-27
30.9. ЗАГРУЗКА СЦЕН В UM SIMULATION	1-28

1. Подготовка и использование фотореалистичного окружения с помощью модуля UM Scene

1.1. Введение

Модуль **UM Scene** представляет собой базу 3D моделей, текстур, фоновых рисунков и набор дополнительных инструментов интегрированных в программу **UM Simulation**.

База 3D моделей включает в себя модели зданий (коттеджи, многоквартирные дома, промышленные здания), автомобильных дорог (прямые участки пути, повороты, перекрёстки), дорожной и уличной инфраструктуры (заборы, скамейки, урны, канализационные люки, дорожные знаки, светофоры, фонари), растений (деревья и кустарники), кузовов популярных автомобилей. Некоторые из моделей представлены в таблице ниже.



Все 3D модели, входящие в поставку "Универсального механизма" лицензированы для коммерческого использования. Любой пользователь "Универсального механизма" может использовать эти 3D объекты произвольным образом: создавать сцены и модели с использованием этих 3D моделей, записывать и распространять полученные с этими сценами видео-файлы и т.д.

1.2. Начинаем работать

В этом разделе приведены инструкции, которые помогут вам быстро ознакомиться с возможностями модуля **UM Scene** и посмотреть на разработанные сцены из примеров. Чтобы начать работать с модулем **UM Scene**, нужно сделать следующие шаги.

- **Установите базу стандартных 3D моделей**, подробнее о базе стандартных объектов см. п. 1.3. "*Структура базы 3D моделей*", стр. 1-5. Вот прямая ссылка на установку этого пакета:

<http://www.universalmechanism.com/download/umscenecollection.exe>

Ссылку на последнюю версию базы стандартных 3D моделей вы всегда можете найти в разделе в разделе "*Универсальный механизм / Инсталляции*" на странице <http://www.universalmechanism.com/pages/index.php?id=3>.

- **Убедитесь, что у вас есть лицензия на модуль UM Scene.** В Универсальном механизме каждый модуль лицензируется отдельно. Поэтому при работоспособной на первый взгляд программе модели со сценами могут не загружаться. Для проверки наличия лицензии запустите программу **UM Simulation**, в главном меню выберите пункт меню **Помощь | О программе** и убедитесь, что напротив модуля **UM Scene** стоит "+", см. рис. 1.1. Если вы недавно установили Универсальный механизм и у вас еще действует пробная лицензия, то у вас этот модуль будет доступен. Пробная лицензия включает все выпущенные модули. Если вы зарегистрированный пользователь УМ обратитесь к нам по адресу um@umlab.ru и мы выдадим вам бесплатную пробную лицензию на три месяца.

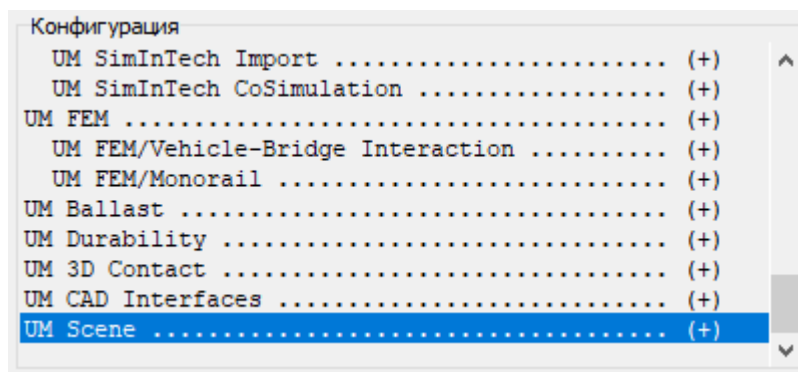


Рис. 1.1. Модуль **UM Scene** в конфигурации УМ

- **Загрузите модели с примерами сцен.** В инсталляцию УМ включено несколько моделей с заранее настроенными конфигурациями, включающими сцены.
 - Модель автотрисы **AC4** включает две конфигурации со сценами: [**Scene - Railroad through village**] и [**Scene - Railroad through forest**]. Запустите программу **UM Simulation**, загрузите модель из каталога {*Данные УМ*} \ *Samples* \ *Rail_Vehicles* \ *ac4*, далее с помощью пункта меню **Файл | Загрузить конфигурацию** загрузите нужные конфигурации, см. рис. 1.2.

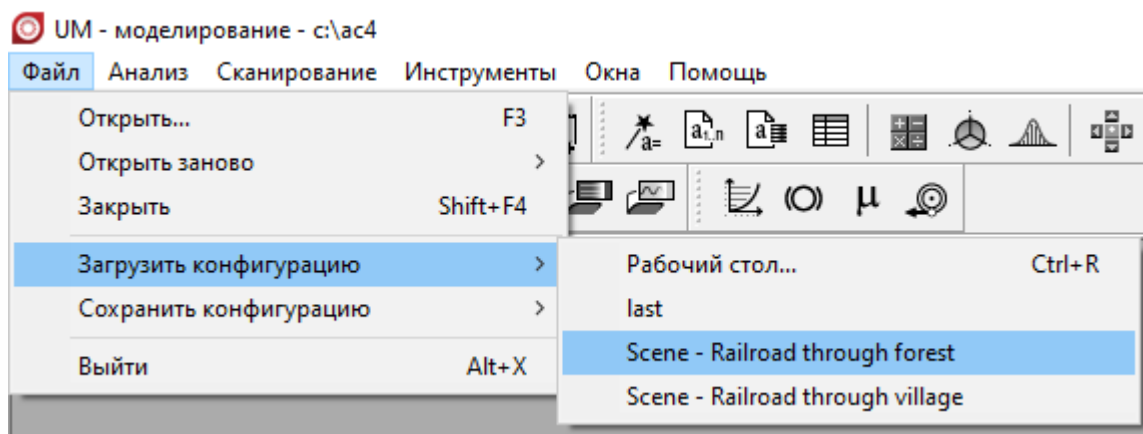


Рис. 1.2. Конфигурации модели AC4

- Модель грузовика с прицепом из каталога $\{Данные\ UM\}\Samples\Automotive\trucktrailer$ включает две предварительно настроенные конфигурации со сценами: [Industrial zone Trajectory 1] и [City zone Trajectory 2].

1.3. Структура базы 3D моделей

База 3D моделей состоит из двух частей: 3D моделей, поставляемых вместе с ПК "Универсальный механизм" (далее "*стандартные 3D модели*") и пользовательской базы 3D моделей. Пользовательская база 3D моделей после инсталляции пустая и наполняется пользователем самостоятельно.

Замечание База стандартных 3D моделей является составной частью ПК "Универсальный механизм", но в силу своего большого объема не входит в инсталляцию ПК "Универсальный механизм". Эту базу нужно скачивать и устанавливать отдельно. Последнюю версию базы 3D моделей вы можете найти на страничке <http://www.universalmechanism.com/pages/index.php?id=3> в разделе *Универсальный механизм / Инсталляции*.

База стандартных 3D моделей по умолчанию устанавливается в рабочий каталог $\{Данные\ UM\}\Scene\ data\Standard\ objects$ и включает в себя ZIP-архив с 3D моделями [*StandardObjects.db*] и XML-файл с описанием моделей из этой базы с их начальным положением, ориентацией и масштабом [*StandardObjects.xml*]. Подробнее понятие рабочего каталога описано в п. 1.1.1 "*Структура размещения UM*", стр. 1-3.

В каталоге $\{Данные\ UM\}\Scene\ data\Scenes$ находятся некоторые заранее разработанные сцены.

Ссылка на последнюю загруженную в модель сцену сохраняется в конфигурационном файле модели *.icf в структуре "*with environment*", поле "*scenefilename*". При загрузке модели поиск указанного в конфигурации файла сцены *.scene выполняется сначала по прямому пути, потом в каталоге модели и затем уже в каталоге $\{Данные\ UM\}\Scene\ data\Scenes$. Учитывайте эту особенность работы программы при копировании ваших моделей и переносе их с компьютера на компьютер.

1.4. Взаимодействие сцен с моделями УМ

Сцены, построенные с помощью модуля **UM Scene**, которые обсуждаются в этой главе, никак **НЕ** взаимодействуют с динамическими моделями: железнодорожный экипаж проедет сквозь дом, а автомобиль сквозь дерево, если такие элементы сцены окажутся на их пути. При разработке сцен для автомобилей или железнодорожных экипажей нужно учитывать это обстоятельство и создавать совместимые сцены, где автомобиль будет ехать по дороге и не проезжать через здания и т.п.

Если вам нужно смоделировать движение транспортного средства в некоторой конкретной сцене, тогда вам нужно правильно расположить сцену, так чтобы корректно задать начальное положение транспортного средства на сцене, и отдельно описать траекторию движения транспортного средства, чтобы она корректно прошла подготовленной сцене. Задание сдвига и поворота сцены выполняется с помощью узла **<SceneCenter>** в настройках сцены, см. узел **<EnvironmentSettings>**. Задание траектории для движения автомобиля описывается в п. 12.3.1. "*Макропрофиль пути*", а для железнодорожных экипажей – в п. 8.3. "*Путь*".

Если для вас первична заданная траектория движения автомобиля или ж.-д. экипажа, тогда вам нужно разработать новую или модифицировать существующую сцену таким образом, чтобы необходимая траектория движения транспортного средства логично и естественно укладывалась бы в эту сцену.

Описание сцены и задание траектории движения транспортного средства – это пересекающиеся задачи, которые нужно решать по отдельности. Например, если вы разработали сцену с улицами, то программа не сможет автоматически подготовить корректную траекторию для езды по ним. И наоборот, если у вас есть траектория, то программа не может автоматически сгенерировать сцену, подходящую для этой траектории.

Также отметим следующее. При моделировании динамики автомобилей и ж.-д. экипажей в УМ их начальное положение всегда совпадает с началом отсчёта глобальной системы координат. Если вам нужно в одной и той же сцене проехать по двум разным траекториям, то для каждого эксперимента нужно соответствующим образом сдвигать и поворачивать сцену с помощью узлов **<EnvironmentSettings>** и **<SceneCenter>**, см. рис. 1.3 .

Если требуется организовать взаимодействие между моделируемой механической системой и элементами сцены, например, столкновение автомобиля с деревом, то соответствующее взаимодействие нужно описывать обычными средствами УМ. Так, например, столкновение автомобиля с деревом можно смоделировать с помощью инструментария из модуля **UM 3D Contact** – связать со стволом дерева (на самом деле с базовым телом **Base0** в месте, где растёт дерево) контактное многообразие в виде цилиндра, а с кузовом автомобиля связать другое контактное многообразие, соответствующее форме кузова, и таким образом описать механическое взаимодействие.

Еще раз отметим, что модуль **UM Scene** не включает никакого расчёта механики взаимодействия и предназначен только для расширенных возможностей по визуализации городских сцен, ландшафтов и т.д. Вместе с тем, элементы сцены обнаруживаются лучевыми сенсорами и расстояние до таких объектов возвращается как результат работы лучевых сенсоров. Описание особенностей взаимодействия сенсоров с объектами сцены дано в 33-ей главе руководства пользователя.

1.5. Положение и ориентация объектов сцены

Положение объектов описывается узлом XML-разметки **<Position>** и задаются в метрах, см. пример ниже.

```
<Position X="25" Y="0" Z="0"/>
```

Ориентация объектов задается с помощью узла XML-разметки **<Rotation>** и вводится тремя различными способами. Углы поворота задаются в градусах.

```
<Rotation Type="rtAngle" Z="90"/>
```

```
<Rotation Type="rtAngles">
```

```
  <Angle Axis="Z" Value="90"/>
```

```
</Rotation>
```

```
<Rotation Type="rtQuaternion" X="0" Y="0" Z="0.70710678" W="0.70710678"/>
```

Ниже рассмотрим способы задания ориентации объектов подробнее.

- **Type="rtAngle"**. Задаёт поворот вокруг одной из осей: Z, X или Y. Если при описании указано несколько осей, то применяться будет только одна из них в соответствии со приоритетом Z, X, Y и неважно в каком порядке эти атрибуты были указаны в XML-разметке, использован будет только один. Дело в том, что стандартные 3D модели заранее приведены в принятую в УМ систему координат, где ось Z направлена вверх, а X и Y образуют горизонтальную плоскость. Таким образом, для построения большинства сцен для каждого объекта обычно достаточно указать только дополнительный угол поворота вокруг вертикальной оси – оси Z. И проще всего это сделать с помощью записи данного типа.
- **Type="rtAngles"**. Ориентация объектов задаётся последовательностью поворотов на произвольный угол вокруг произвольных осей. Этот тип описания ориентации объекта практически тождественен способу описания ориентации графических элементов в системе координат графических объектов, принятому в УМ и позволяет задать любую необходимую сколь угодно сложную ориентацию объекта.
- **Type="rtQuaternion"**. Задание ориентации объекта с помощью кватерниона¹. Этот тип описания ориентации используется по умолчанию. Если **Type="rtQuaternion"**, то его можно не указывать.

Узлы описания XML-разметки **<Position>** и **<Rotation>** можно опускать и, соответственно, не вводить дополнительный сдвиг или поворот объекта.

1.6. Десятичный разделитель и регистр имен узлов и атрибутов

Как и в остальном "Универсальном механизме" в качестве десятичного разделителя должна использоваться точка.

Имена узлов и атрибутов чувствительны к регистру символов. Корректные формы приведены в файлах сцен из примеров. Рассмотрим, например, атрибут **Name**. Корректной

¹ Подробнее о кватернионах см., например, по ссылке

<https://ru.wikipedia.org/wiki/%D0%9A%D0%B2%D0%B0%D1%82%D0%B5%D1%80%D0%BD%D0%B8%D0%BE%D0%BD>

формой является **Name**; формы **NAME**, **name** – не поддерживаются и приведут к ошибке чтения файла(ов).

1.7. Создание сцен

Сцены сохраняются в XML-файлы с расширением *.scene. Эти файлы имеют типовую XML-разметку. Синтаксис XML-разметки не будет детально разбираться в этом документе. Об этом можно прочесть в большом количестве источников, в частности здесь: <https://ru.wikipedia.org/wiki/XML>.

В настоящей версии модуля **UM Scene** создание файлов *.scene возможно только вручную с помощью текстового редактора. Полноценный интерактивный редактор сцен будет выпущен в одном из следующих релизов.

1.7.1. Формат файла сцены

Ниже рассмотрим основные узлы файла описания сцен *.scene и их атрибуты. Общая структура файла сцены представлена на рис. 1.3.

Корневой узел <Scene> включает узлы <EnvironmentSettings> и <Objects>. В узле <EnvironmentSettings> содержится описание скайбокса² (узел <SkyBox>) и положение системы координат сцены в глобальной системе координат модели. Положение и ориентация всех объектов в сцене задается в системе координат сцены. Изменение положения или ориентации системы координат сцены относительно глобальной системы координат модели приведёт к сдвигу/повороту всех объектов сцены.

В узле <Objects> содержится список узлов <Object>, которые описывают объекты сцены из стандартной и пользовательской базы 3D моделей с их положением, ориентацией и масштабом.

Типовая сцена включает описание скайбокса, подложки (underlay) и набора объектов сцены. Как было сказано выше, некоторые заранее разработанные сцены поставляются вместе с "Универсальным механизмом" в каталоге {Данные UM}\Scene data\Scenes.

Узел <Object> описывает объект на сцене. Рассмотрим структуру этого узла.

- **ID** – уникальный идентификатор объекта из стандартной или пользовательской базы объектов. Соответствует полю **ID** из файлов описания стандартных (файл *StandardObjects.xml*) или пользовательских (файл *UserObjects.xml*) объектов сцены.
- **Name** – произвольное имя объекта, **уникальное!** в контексте описания данной сцены.
- **Position** – положение объекта в системе координат сцены (м).
- **Rotation** – ориентация объекта в системе координат сцены. Подробнее о формате задания ориентации см. п. 1.5 "Положение и ориентация объектов сцены", стр. 1-7.
- **Scale/Value** – масштабный множитель для объекта из базы. Чтобы сделать объект в 10 раз больше – установите коэффициент 10; чтобы сделать объект в 10 раз меньше – используйте коэффициент 1/10.

² Скайбóкс (англ. sky — «небо» и box — «коробка») — объект в трёхмерной графике, играющий роль неба и горизонта, подробнее см.

<https://ru.wikipedia.org/wiki/%D0%A1%D0%BA%D0%B0%D0%B9%D0%B1%D0%BE%D0%BA%D1%81>





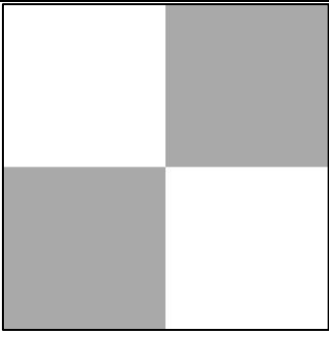
Замечание Поле **Name** обязательно должно быть уникальным в контексте сцены. При загрузке такой сцены будут обнаружены ошибки и сформировано соответствующее сообщение пользователю.

```
<?xml version="1.0" encoding="utf-8"?>
<Scene Description="Sample scene">
  <EnvironmentSettings>
    <SkyBox Enabled="true" Material="SkyBox/CloudyNoon"/>
    <SceneCenter>
      <Position X="10" Y="0" Z="0"/>
      <Rotation X="0" Y="0" Z="0" W="1"/>
    </SceneCenter>
  </EnvironmentSettings>
  <Objects>
    <Object Name="underlay1" ID="1100005">
      <Position X="0" Y="0" Z="-0.1"/>
      <Rotation X="0" Y="0" Z="0" W="1"/>
      <Scale Value="1"/>
      <Dimensions Width="100" Height="100"/>
    </Object>
    <Object Name="suburbanhouse_3" ID="1200004">
      <Position X="50" Y="0" Z="0"/>
      <Rotation Type="rtAngle" Z="90"/>
      <Scale Value="1"/>
    </Object>
    <Object Name="skoda_yeti_1" ID="2100080">
      <Position X="0" Y="0" Z="0"/>
      <Rotation X="0" Y="0" Z="0" W="1"/>
      <Scale Value="1"/>
    </Object>
  </Objects>
</Scene>
```

Рис. 1.3. Структура файла сцены


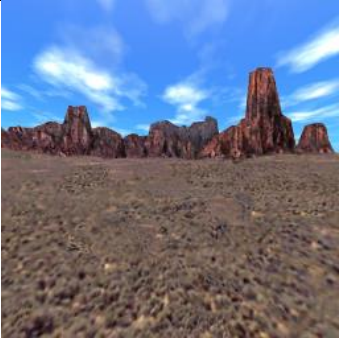
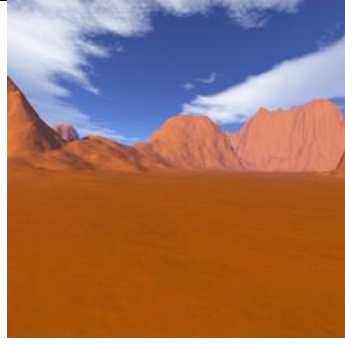
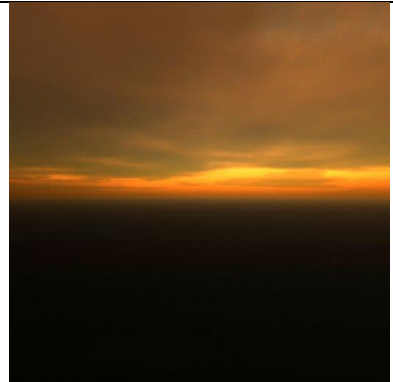
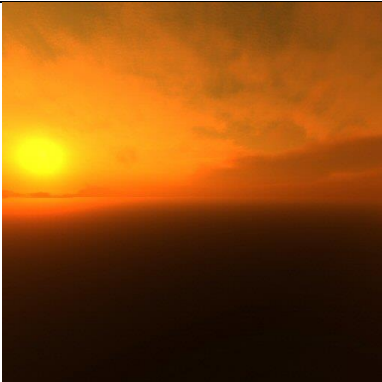



1.7.2. Подложки

В базу стандартных 3D моделей входят следующие подложки, указанные в таблице ниже. Используйте ID, указанные в таблице, для выбора нужной подложки в вашей сцене.

		
<p>Пять видов асфальта, ID 1100010-1100014</p>	<p>Шесть видов травы ID 1100001-1100006</p>	<p>Щебень ID 1100016</p>
		
<p>Три вида песка ID 1100007-1100009</p>	<p>Шахматная подложка ID 1100015</p>	

1.7.3. Скайбоксы

В базу стандартных 3D моделей входят следующие скайбоксы.

		
CloudyNoon	Desert1	Desert2
		
EarlyMorning	Evening	IceRiver
		
Yokohama2	Meadow	

Скайбоксы описываются в секции **<EnvironmentSettings>** следующим образом:
`<SkyBox Enabled="true" Material="SkyBox/CloudyNoon"/>`.

В примере выше вместо наименований *CloudyNoon* можно задать любое название скайбокса из таблицы выше.

Скайбокс *CloudyNoon* подойдет для большинства сцен и рекомендуется как скайбокс по умолчанию.

1.7.4. Создание и редактирование файлов сцен

Интерактивный редактор сцен как отдельный инструмент недоступен в первых версиях модуля. Создание и редактирование файлов сцен нужно выполнять в обычном текстовом редакторе.

Для добавления каждого отдельного объекта в сцену вам понадобится добавить следующий блок в узел **<Objects>** файла сцены:

```
<Object Name="suburbanhouse_3" ID="1200004">  
  <Position X="50" Y="0" Z="0"/>  
  <Rotation Type="rtAngle" Z="90"/>  
  <Scale Value="1"/>  
</Object>
```

Описание полей и формата этого узла дано в п. 1.7.1. "*Формат файла сцены*", стр. 1-8. Еще раз напомним, что значения атрибута **Name** должно быть уникальным в контексте файла сцены. При наведении курсора мыши в анимационном окне на этот объект вы увидите имя объекта в строке статуса внизу анимационного окна. Поле **ID** – это уникальный идентификатор объекта из стандартной или пользовательской базы объектов.

Как можно понять, какой **ID** соответствует каждому конкретному объекту из базы стандартных 3D объектов? Для этого мы разработали набор демонстрационных сцен, куда вошли все стандартные 3D объекты. Демонстрационные сцены начинаются с префикса *<demo>* и находится в каталоге *{Данные УМ}\Scene data\Scenes*.

Запустите программу **UM Simulation**, загрузите любую модель, с помощью пункта меню **Инструменты | Сцены** откройте мастер загрузки сцен и последовательно загрузите все демонстрационные сцены для ознакомления.

Чтобы определить **ID** объекта в сцене сделайте следующее. Откройте анимационное окно и в нём наведите мышку на этот объект. В строке статуса вы увидите имя объекта в файле сцены, см. рис. 1.4. Откройте файл сцены в текстовом редакторе и найдите это имя, см. рис. 1.5. Помните, что имена объектов в контексте сцен – уникальные, поэтому такое имя будет только одно. Рядом с именем вы найдёте **ID** объекта и можете его использовать в своей сцене:

```
<Object Name="suburbanhouse3_1" ID="1200004">
```

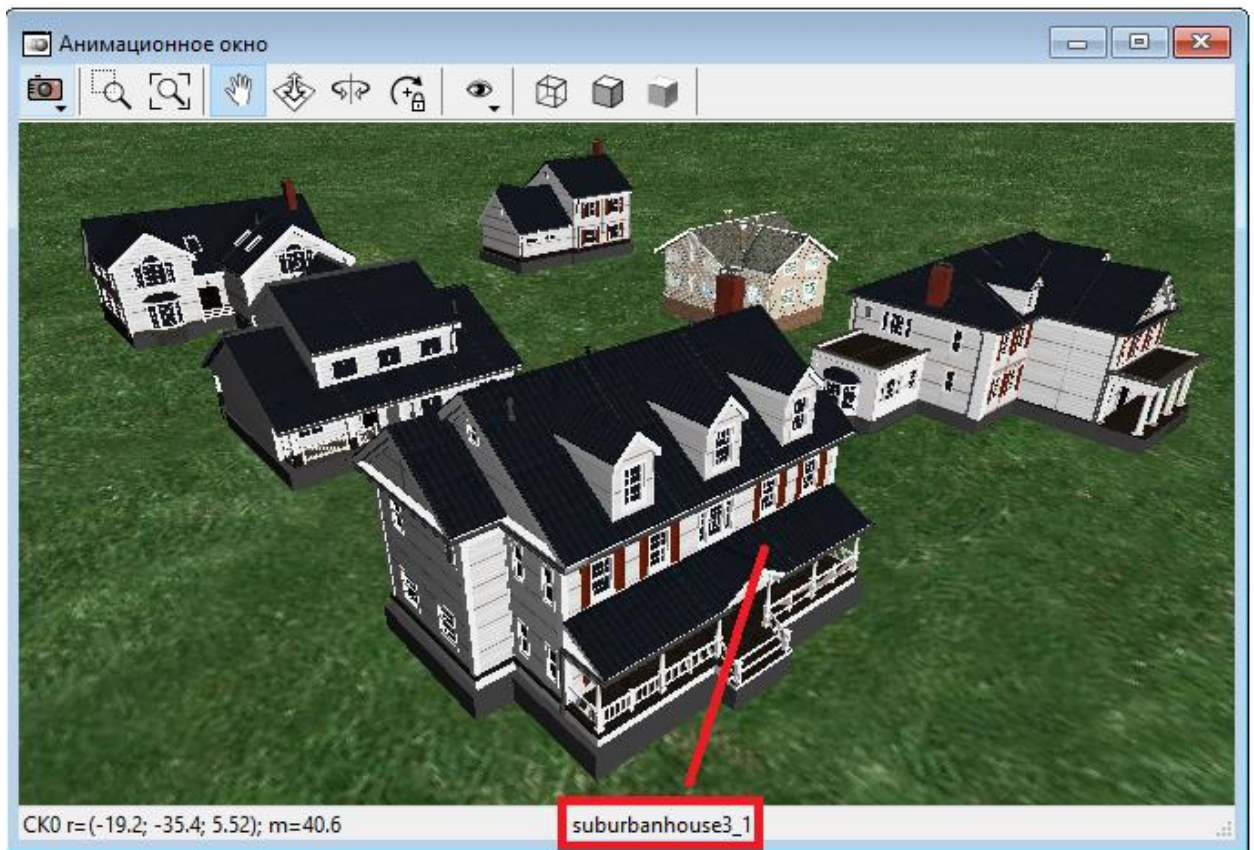


Рис. 1.4. Демонстрационная сцена "demo Suburban house pack.scene"

```

<Object Name="suburbanhouse2_1" ID="1200002">
  <Position X="20" Y="30" Z="0"/>
  <Rotation X="0" Y="0" Z="1" W="0"/>
  <Scale Value="1"/>
</Object>
<Object Name="suburbanhouse3_1" ID="1200003">
  <Position X="-20" Y="-30" Z="0"/>
  <Rotation X="0" Y="0" Z="0" W="1"/>
  <Scale Value="1"/>
</Object>
<Object Name="suburbanhouse4_1" ID="1200004">
  <Position X="-20" Y="30" Z="0"/>
  <Rotation X="0" Y="0" Z="0" W="1"/>
  <Scale Value="1"/>
</Object>
    
```

Рис. 1.5. Демонстрационная сцена "demo Suburban house pack.scene"

1.8. Работа с пользовательской базой 3D объектов

ПК "Универсальный механизм" использует графическую подсистему *Ogre3D*, см. <https://www.ogre3d.org>. Все объекты для пользовательской базы данных должны быть подготовлены в формате *Ogre3D*. *Ogre3D*, как довольно популярное приложение с большой экосистемой, имеет набор инструментов для экспорта графических объектов из ряда популярных программ для 3D-моделирования. Поддерживается экспорт данных из [Autodesk 3ds Max](#), [Blender](#), Maya, COLLADA, MilkShape 3D, Assimp. Список инструментов для экспорта доступен на странице <http://wiki.ogre3d.org/OGRE+Exporters>.

1.8.1. Экспорт объекта из Autodesk 3ds Max

Замечание Для экспорта 3D объектов из **Autodesk 3ds Max** в формате *Ogre3D* установите [Easy Ogre Exporter](#).

Рассмотрим пример добавления объекта в пользовательскую базу 3D объектов из **Autodesk 3ds Max** на модели автомобиля **Skoda Yeti**. Эту модель можно найти в каталоге *{Данные УМ}\Scene data\Samples\Skoda Yeti*.

1. Запустите **Autodesk 3ds Max**. Загрузите модель автомобиля с помощью пункта меню **File | Open** для файлов *.max или **File | Open | Import | Import** для других типов файлов.

Замечание Копии экрана в настоящем руководстве приведены для **Autodesk 3ds Max 2018**. Для других версии **Autodesk 3ds Max** интерфейс программы может отличаться.

Дальнейшая работа с объектом будет проходить с помощью диалогового окна **Scene Explorer**, см. рис. 1.6. Обычно оно находится в левой части окна проекций. Если Вы его не обнаружили, тогда откройте окно через пункт меню **Scene | Manage Scene Content | Scene Explorer** или вызовите его комбинацией клавиш **Alt+Ctrl+O**.

В окне **Scene Explorer** отображаются объекты сцены. В нашем случае сцена содержит в себе 4 объекта: **front_left**, **front_right**, **front_types** и **skoda_yeti**, см. рис. 1.6. При нажатии на каждый отдельный объект левой кнопкой мыши этот объект будет выделяться в проекционном окне.

Если нужно экспортировать сцену одним объектом, то все элементы сцены необходимо объединить в один объект. Этот случай рассмотрен ниже в п. 1.8.1.1. "Объединение", стр. 1-15. Если же в результате экспорта требуется получить каждый объект как отдельный элемент сцены, то объединять компоненты сцены не нужно и можно сразу перейти к экспорту, см. п. 1.8.1.2. "Экспорт объектов", стр. 1-18.

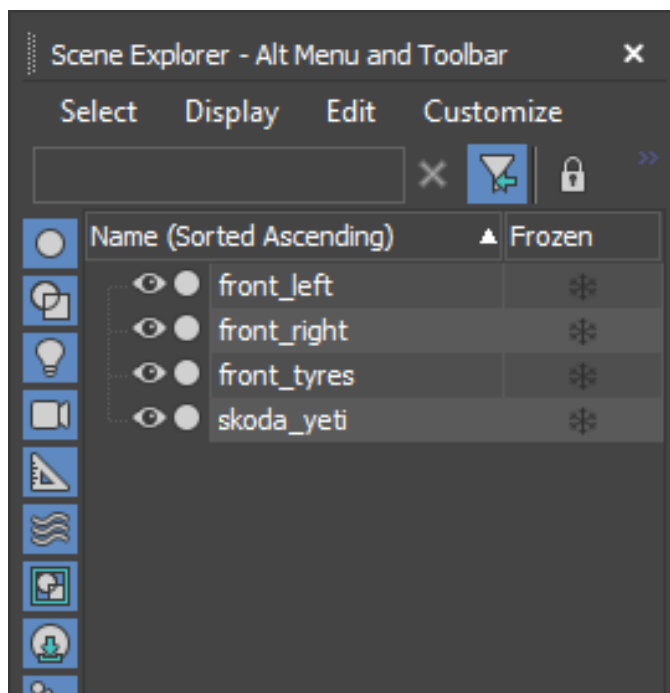


Рис. 1.6 Диалоговое окно Scene Explorer

1.8.1.1. Объединение объектов

Существует несколько способов объединения объектов в **3D Studio Max**. Одним из них является объединение объектов на полигональном уровне. Для этого необходимо выполнить следующие шаги.

1. В окне **Scene Explorer** нужно выбрать элемент, к которому требуется присоединить другой объект. Для этого нажмите на него левой кнопкой мыши, либо выберите элемент прямо в проекционном окне с помощью левой кнопки мыши. Выбранный элемент должен подсветиться. Для модели **Skoda_Yeti** выберите, например, объект **front_left**, см. рис. 1.7.

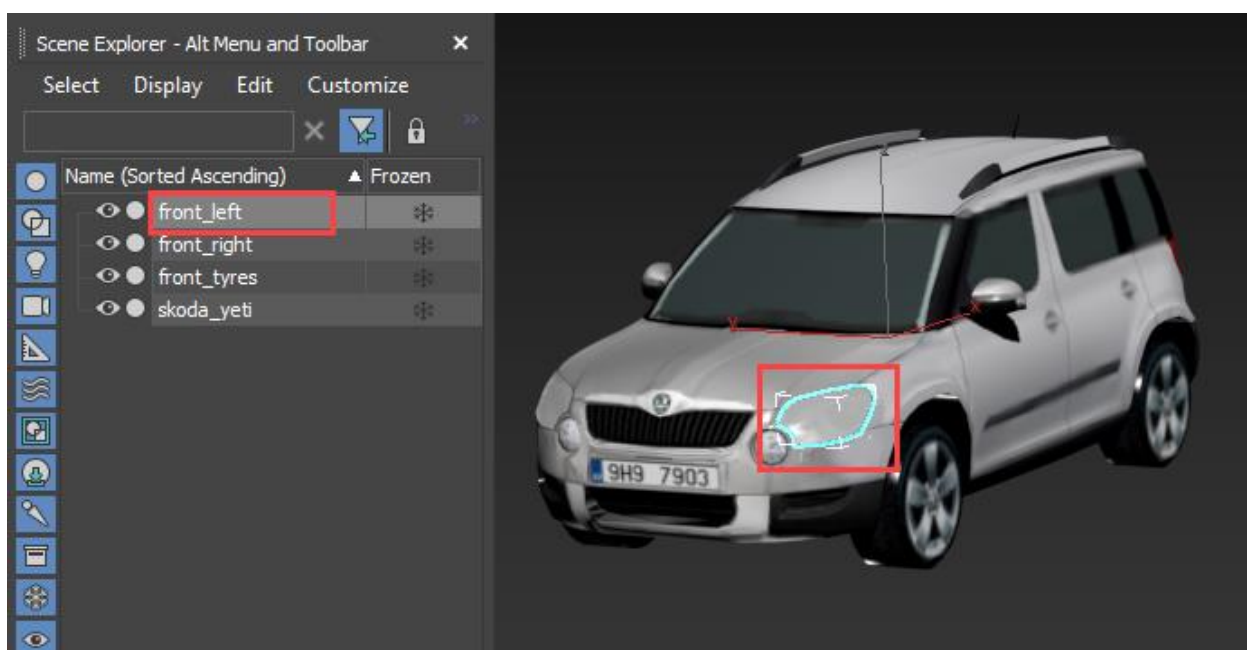


Рис. 1.7 Выбор элемента для объединения

2. После этого выделенный объект необходимо конвертировать в редактируемый полигон. Для этого в проекционном окне на выделенном объекте щелкните правой кнопкой мыши и в контекстном меню выберите **Convert to | Convert to EditablePoly**, см. рис. 1.8.

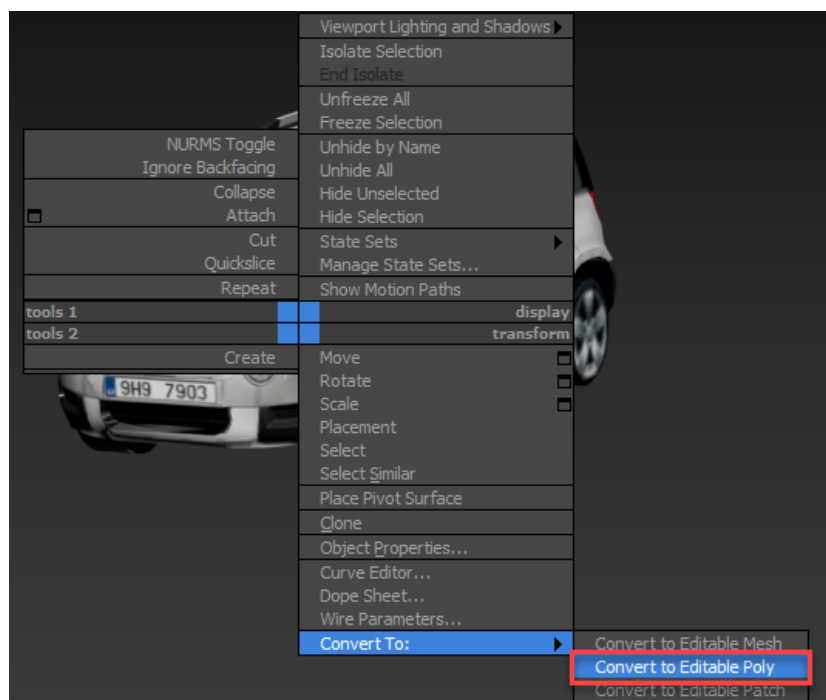


Рис. 1.8 Конвертация объекта в редактируемый полигон

3. После этого необходимо снова щелкнуть правой кнопкой мыши по выделенному объекту и в контекстном меню выбрать пункт **Attach**, см. рис. 1.9.

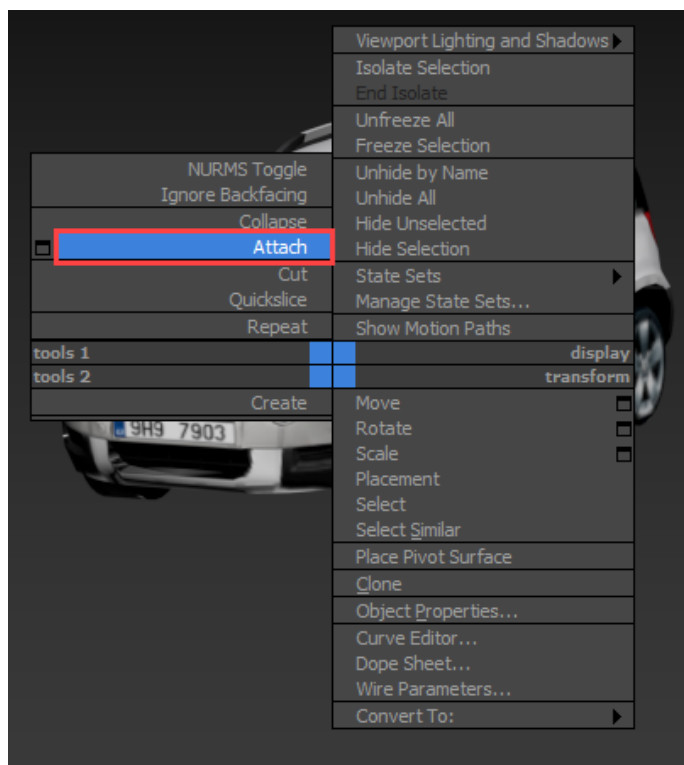


Рис. 1.9 Подготовка к объединению объектов через пункт меню **Attach**

4. Далее перейдём к объединению объектов. На объект, который хотите присоединить, нажмите левой кнопкой мыши либо в проекционном окне, либо в окне **Scene Explorer**. Для нашей модели выберите, например, объект **front_right**. После этого в окне **Scene Explorer** останется уже только 3 объекта, так как объект **front_right** присоединился к объекту **front_left**, а в проекционном окне будут подсвечены обе фары, см. рис. 1.10.

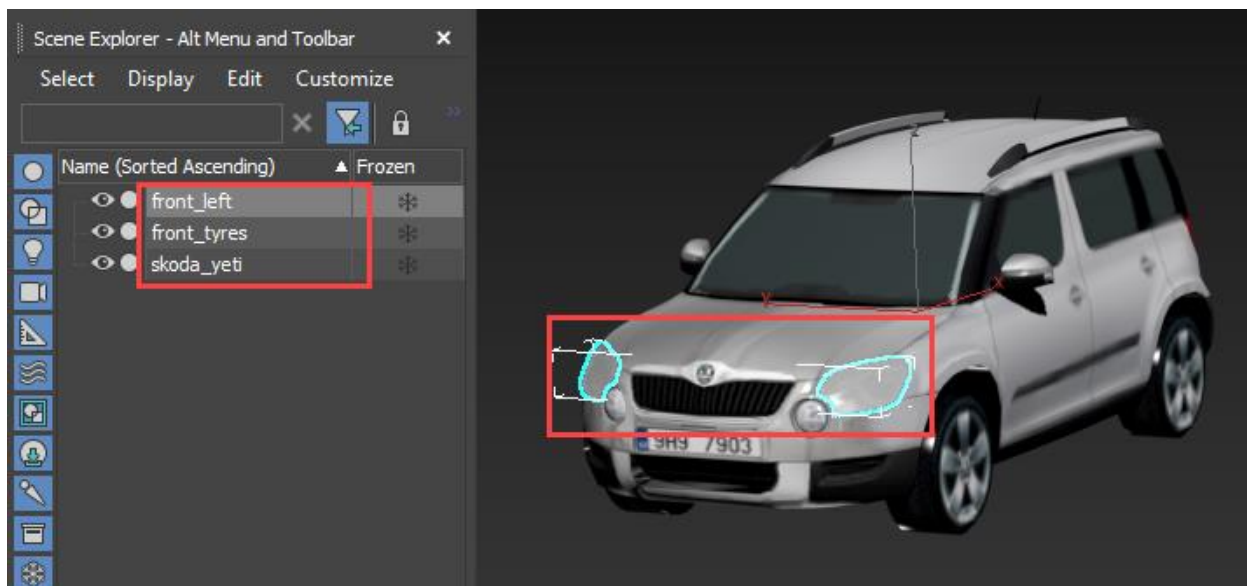


Рис. 1.10 Изображение объединённых фар

5. Теперь выберите объект **front_tyres**. В проекционном окне появится всплывающее окно **Attach Options**. Это связано с тем, что присоединяемый объект имеет материал, не совпадающий с материалом объекта, к которому он присоединяется. В случае появления данного окна выберите пункт **Match Material IDs to Material** и поставьте галочку **Condense Material and IDs** и нажмите кнопку **OK**, см. рис. 1.11. После этого объект автоматически присоединится.

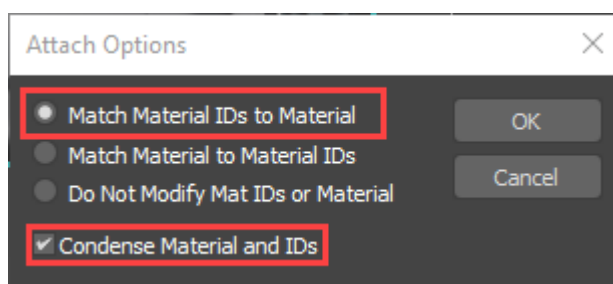


Рис. 1.11 Диалоговое окно **Attach Options**

6. Присоедините последний объект **skoda_yeti** к объекту **front_left**, как описано в п. 5. В итоге в окне **Scene Explorer** должен остаться один объект **front_left**, а в проекционном окне будет подсвечен весь автомобиль, как показано на рис. 1.12.

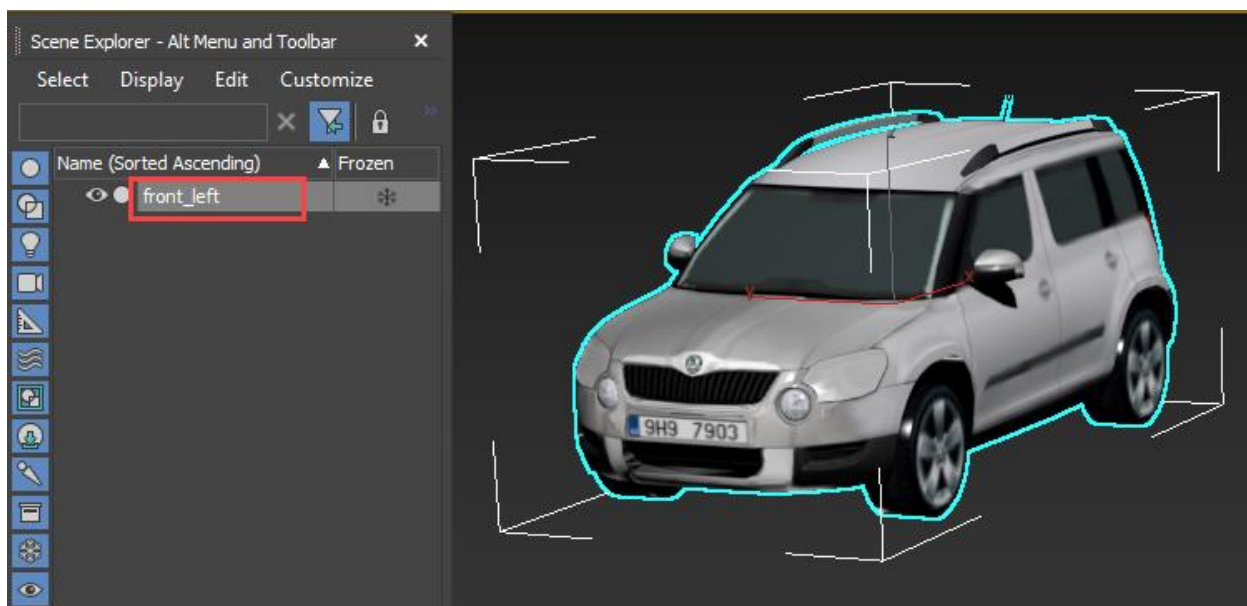


Рис. 1.12 Изображение объединенного объекта

Замечание Если требуется объединять другие объекты, не взаимодействующие с уже объединенными, нажмите на пустом месте проекционного окна сначала правой кнопкой мыши, затем левой. Параметр **Attach** для выбранного объекта будет сброшен. После этого повторите п.1-5.

1.8.1.2. Экспорт объектов

Для экспорта объекта или группы объектов требуется выполнить следующие шаги.

1. Если требуется экспортировать все элементы сцены, то нужно выбрать пункт главного меню **File | Export | Export...** . Если требуется экспортировать только некоторые избранные элементы, то нужно выбрать эти элементы в **Scene Explorer** и выполнить пункт главного меню **File | Export | Export Selected....** В нашем случае выберите экспорт, как показано на рис. 1.13.

2. В появившемся окне **Select File to Export** выберите путь, куда будете сохранять экспортированные файлы. После этого создайте папку с названием экспортируемого объекта. Из этой папки будет формироваться наполнение пользовательской базы. Название должно быть обязательно написано латиницей. Для нашей модели создайте папку **Skoda_Yeti** и откройте ее, см. рис. 1.14.

3. В окне **Select File to Export** в поле **File name** задайте имя экспортируемому файлу (на латинице без пробелов). Это имя будет установлено для файла с информацией об экспортированных объектах. Для нашей модели введите имя **skoda_yeti**, см. рис. 1.14.

4. В поле **Save as type** выберите формат **Ogre Scene (*.SCENE)** и нажмите кнопку **Save**, см. рис. 1.14.

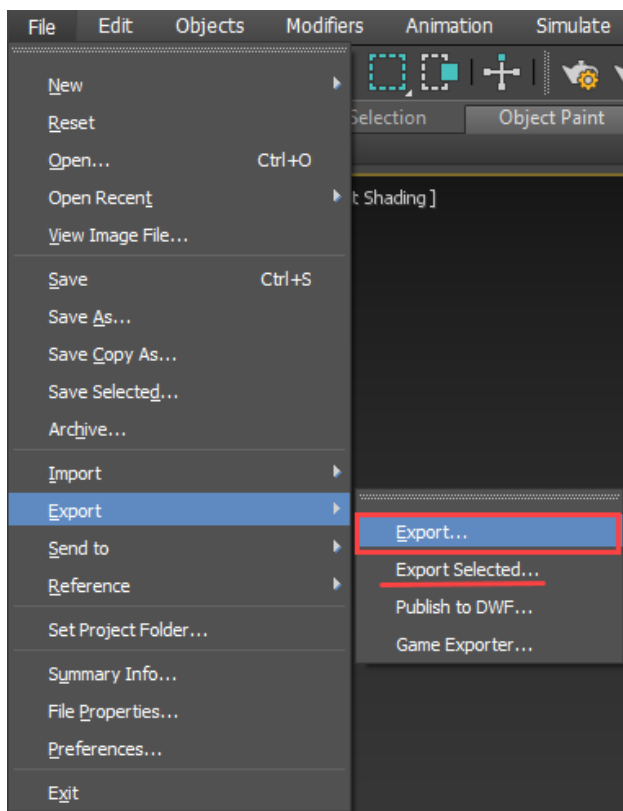


Рис. 1.13 Экспорт объекта

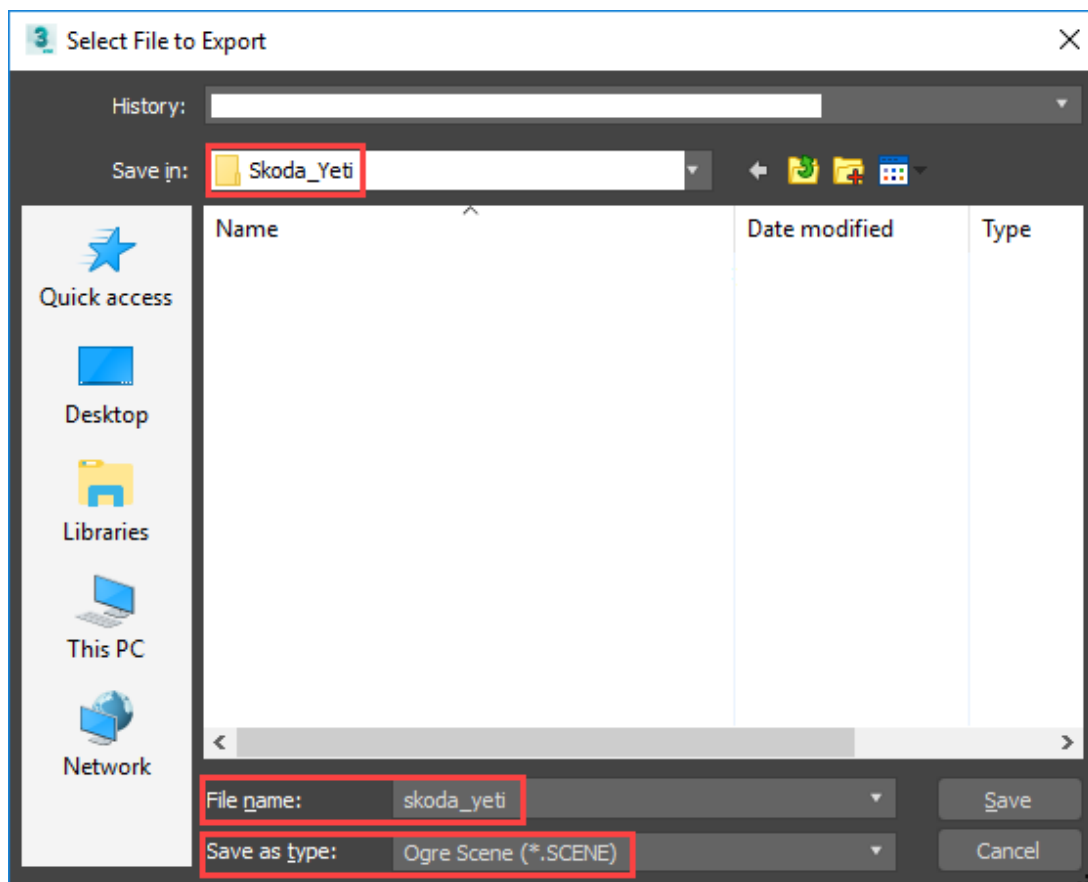


Рис. 1.14 Создание папки и выбор типа при экспорте

5. После этого в проекционном окне появится окно **Easy Ogre Exporter** см. рис. 1.15. В разделе **Basic configuration** в поле **Ogre version** выберите **Ogre 1.8**. Для текущей версии модуля **UM Scene** поддерживаются только версия **Ogre 1.8!** В поле **Material sub dir** отображается название папки, в которую будет сохранен файл материала с таким же названием, как была названа папка для сохранения, и с расширением **.material**. В поле **Texture sub dir** указывается название папки, в которую будут сохраняться файлы текстур объекта. В поле **Mesh sub dir** указано название папки, в которой будут расположены файлы с именами экспортируемых объектов с расширением **.mesh**.

6. В разделе **Materials** в поле **Pixel lighting (CG)** выберите **All materials (3 lights)**. Это нужно для учета материалом всего светового спектра.

7. Все остальные настройки установите, как показано на рис. 1.15 и нажмите **ОК**.

8. При успешном экспорте файлов появится окно **Info**, см. рис. 1.16.

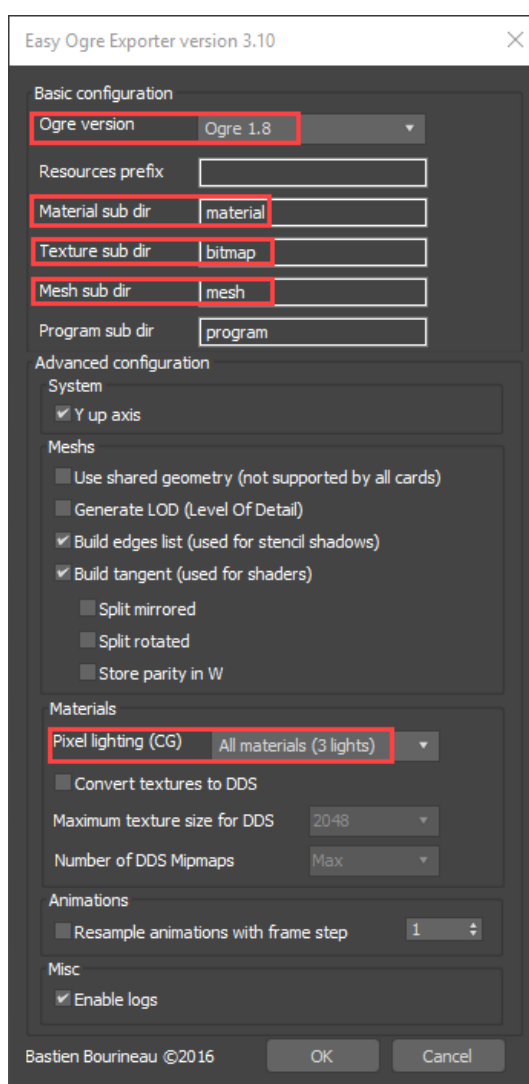
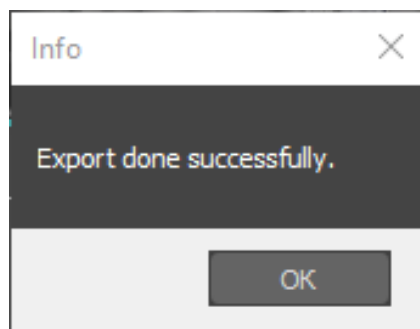


Рис. 1.15 Диалоговое окно **Easy Ogre Exporter**

Рис. 1.16 Диалоговое окно **Info**

1.8.1.3. Типичные проблемы экспорта и их исправление

В результате экспорта некоторые файлы могут быть экспортированы некорректно. Рассмотрим несколько типовых проблем и покажем, как их преодолеть.

- При экспорте объектов не всегда экспортируются файлы текстур. В таком случае их необходимо вручную переносить из исходной папки с объектом в папку, предназначенную для хранения экспортированных файлов. В нашем примере с моделью Skoda Yeti не был экспортирован файл текстуры. Зайдите в исходную папку и скопируйте файл `diffuse_white.jpg` в папку `Skoda_Yeti`, см. п. 1.8.2.1. "Создание ZIP-архива", стр. 1-24.
- В большинстве случаев необходимо скорректировать файл с расширением ***.material**. Откройте файл `skoda_yeti.material` в любом текстовом редакторе. В текущей версии модуля **UM Scene** некоторые параметры материалов не поддерживаются. Для корректного отображения объекта в **UM Simulation** нужно установить в ноль первые три числа для полей **ambient**, **specular** и **emissive** для всех материалов – **0 0 0 1**, см. рис. 1.17

```

material "Material_32"
{
    technique Material_32_technique
    {
        pass Material_32_standard
        {
            ambient 1 1 1 1
            diffuse 1 1 1 1
            specular 0 0 0 0 25.5
            emissive 0 0 0 1
            vertex_program_ref vsLightGENO
            {
            }
            fragment_program_ref fpLightGENDIFF0
            {
            }

            texture_unit Material_32_Diffuse#0
            {
                texture diffuse_white.jpg
                tex_coord_set 0
                colour_op modulate
            }
        }
    }
}

```

```

material "Material_32"
{
    technique Material_32_technique
    {
        pass Material_32_standard
        {
            ambient 0 0 0 1
            diffuse 1 1 1 1
            specular 0 0 0 1 25.5
            emissive 0 0 0 1
            vertex_program_ref vsLightGENO
            {
            }
            fragment_program_ref fpLightGENDIFF0
            {
            }

            texture_unit Material_32_Diffuse#0
            {
                texture diffuse_white.jpg
                tex_coord_set 0
                colour_op modulate
            }
        }
    }
}

```

Рис. 1.17 Описание свойств материалов

- Еще одной распространенной проблемой при экспорте является несовпадение названия или расширения текстурного файла и его описания в файле материала. В нашем случае текстура **diffuse_white.jpg** была экспортирована корректно, см. Рис. 1.17.
- Иногда требуется сделать объект прозрачным. Для этого в после описания материалов нужно добавить две строки, отмеченные на рис. 1.18. Последнее (четвертое) число в поле **diffuse** отвечает за прозрачность объекта. Прозрачность устанавливается в интервале от 0 до 1, где 0 соответствует полностью прозрачному материалу, а 1 – полностью непрозрачному.

Добавьте следующие свойства материалов для добавления прозрачности:

scene_blend alpha_blend

depth_write off

```
material "Material__31_20000001"
{
    technique Material__31_technique
    {
        pass Material__31_standard
        {
            ambient 0 0 0 1
            diffuse 0.533333 0.533333 0.533333 0.3
            specular 0 0 0 1 0
            emissive 0 0 0 1
            scene_blend alpha_blend
            depth_write off
        }
    }
}
```

Рис. 1.18 Добавление прозрачности объекту

- Для файла с расширением *.material необходима еще одна обязательная коррекция. В файле в поле **texture** необходимо указывать полный путь к используемой текстуре с применением прямого слеша, начиная с последующего каталога после созданного zip-архива. Для нашего примера для материала с названием **Material__32_20000001** в строке чтения текстуры перед ее названием добавьте **Skoda_Yeti/**, см. рис. 1.19.

```
material "Material__32_20000001"  
{  
    technique Material__32_technique  
    {  
        pass Material__32_standard  
        {  
            ambient 0 0 0 1  
            diffuse 1 1 1 1  
            specular 0 0 0 1 25.5  
            emissive 0 0 0 1  
  
            texture_unit Material__32_Diffuse#0  
            {  
                texture Skoda_Yeti/diffuse_white_20000001.jpg  
                tex_coord_set 0  
                colour_op modulate  
            }  
        }  
    }  
}
```

Рис. 1.19 Описание полного пути к текстуре

Замечание Обратите внимание на то, что в файлах с расширением *.material имена используемых материалов не должны повторяться с именами данного и других объектов, иначе это приведет к ошибке. То есть, название материала, например, Material__32_20000001, больше нигде использоваться не может. Так же сами файлы с расширением *.material должны носить различные имена.

1.8.2. Создание пользовательской базы объектов

База пользовательских объектов состоит из **XML-файла** с описанием структуры базы и **ZIP-архива** с данными. ZIP-архив содержит экспортируемые файлы, а XML-файл – данные об экспортируемом объекте. Рассмотрим создание каждого из этих файлов.

Замечание Для вашего удобства предварительно подготовленные ZIP- и XML-файлы находятся в каталоге `[{UM Data}\Scene data\Samples\User objects]`.

1.8.2.1. Создание ZIP-архива

1. Создайте папку с названием **SampleUserBase**. Имя папки должно быть на латинице без пробелов.
2. После этого в папке **SampleUserBase** создайте папку **Skoda_Yeti_Sample**. Эта папка будет служить для хранения экспортированных файлов рассматриваемой модели.
3. Скопируйте в созданную папку **SampleUserBase\Skoda_Yeti_Sample** файлы с расширением *.mesh, *.material и файлы текстур из папки **skoda_yeti**. Для нашего примера

необходимо скопировать файлы **front_left.mesh**, **skoda_yeti.material** и **diffuse_white.jpg**, см. рис. 1.20.

4. Для удобства переименуйте файл **front_left.mesh** в **skoda_yeti.mesh**.

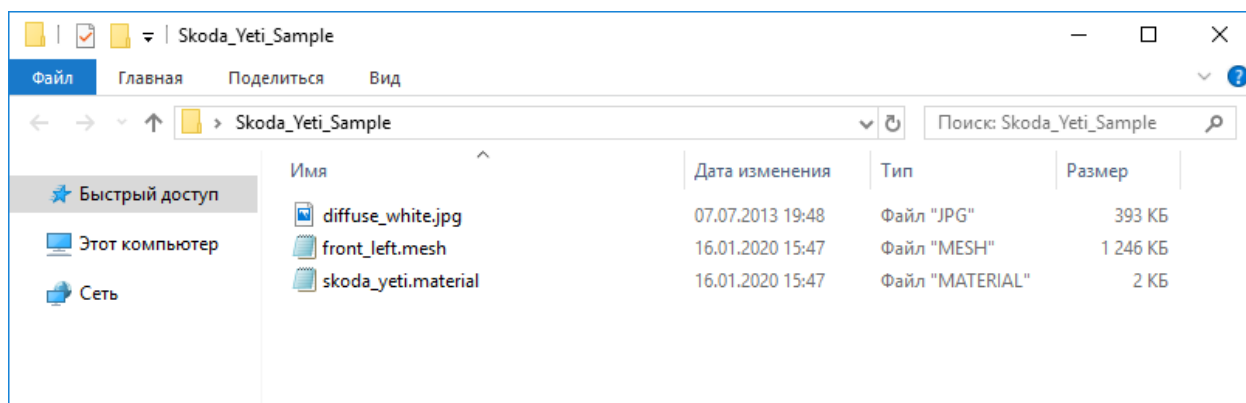


Рис. 1.20. Создание папки для хранения файлов 3D-объекта

5. После устранения ошибок с описанием материала(ов), описанных в п. 1.8.1.3. "Типичные проблемы экспорта и их исправление", стр. 1-21, заархивируйте папку **SampleUserBase** любой программой, предназначенной для упаковки файлов в единый файл-архив. В нашем случае заархивированный файл будет иметь имя **SampleUserBase.zip**.

Важно! Имя архива обязательно должно быть написано латиницей и без пробелов. Расширение файла обязательно должно быть ***.zip**. При архивировании рекомендуется устанавливать максимально возможный уровень сжатия данных.

6. Переместите архив в стандартную папку пользовательской базы *{Данные УМ}\Scene data\User object*, см. рис. 1.21.

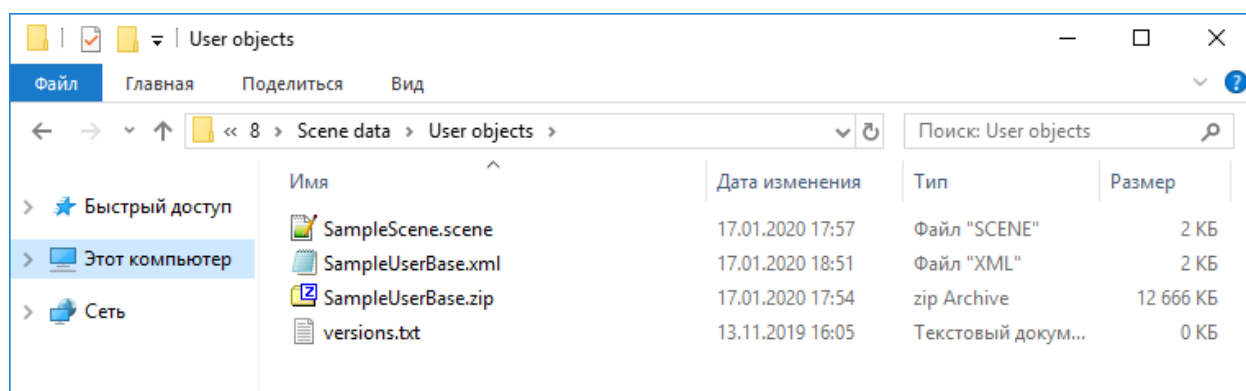


Рис. 1.21 Содержание папки *{Данные УМ}\Scene data\User object*

Замечание Каждый архив должен быть меньше 2 Гб. Если ваши объекты вместе приводят к тому, что размер архива превышает 2 Гб, разбейте объекты на несколько архивов. Размер отдельных неупакованных MESH-файлов также не должен превышать 2 Гб. Загрузка отдельных MESH-файлов больше 2 Гб в неупакованном виде не поддерживается.

1.8.2.2. Создание XML-файла

В текущей версии модуля **UM Scene** XML-файл необходимо создавать вручную с помощью текстового редактора. XML-файл содержит информацию о 3D-моделях, необходимую для отображения объектов на сцене. Для нашего примера с экспортом модели Skoda Yeti назовём этот файл **SampleUserBase.xml**. Ниже на примере некоторых моделей рассмотрим узлы файла и их атрибуты, см. рис. 1.22.

- Главным атрибутом узла **Base** является **FileName**. Этот атрибут содержит имя ZIP-архива, в котором содержатся описываемые объекты.
- Объекты в архиве могут быть разбиты на группы, например "*Комтеджи*", "*Дерево*" и т.д. Для этого используется узел **Group**. Атрибут **ID** узла **Group** задает числовое значение группы, отвечающее за группировку объектов в списках редактора. Рекомендуется указывать уникальное значение для каждого файла пользовательской базы начиная с 20000000. **Name** – произвольное имя узла **Group**. Имя не требует уникальности.

Каждый последующий узел **<Object>** содержит описание загрузки добавляемого объекта, включая в себя начальный сдвиг и поворот, а также масштаб 3D-объекта и материал для данного объекта. Рассмотрим структуру этого узла.

- Атрибут **Name** – произвольное имя, должно быть уникальным для каждого описываемого объекта.
- Атрибут **Description** предназначен для краткого описания объекта.
- Атрибут **ID** – уникальный идентификатор описываемого объекта. Рекомендуется каждый последующий **ID** увеличивать на единицу от предыдущего. Начинать нумерацию нужно от **ID+1**, который описывается в узле **<Group>**. Например, если **ID** группы равен 20000000, то **ID** первого описываемого объекта должен быть 20000001. Обратите внимание, что **ID** объекта должен быть уникальным для всех стандартных и пользовательских баз 3D объектов. Стандартные объекты имеют **ID** меньше 20000000. Таким образом, устанавливайте **ID** для объектов пользовательской базы 3D объектов начиная с 20000000.
- Атрибут **Type** указывает тип добавляемого объекта в большинстве случаев он будет равен **otMesh**. При описании подложки атрибут **Type** устанавливается в значение **otUnderlay**. В этом случае для описания материала используется тип **smtMaterial**, см. ниже.
- Узел **MeshFileName** хранит название файла с расширением *.mesh. В текущей версии, как и для текстур, необходимо указывать полный путь к mesh-файлу с применением прямого слеша, начиная с последующего каталога после созданного zip-архива, см. рис. 1.22.

Узлы **Position** и **Rotation** описывают начальный сдвиг и поворот объекта в системе координат сцены. При экспорте объекта из Autodesk 3ds Max создаётся файл с расширением *.scene. В нем содержатся положение и ориентация объекта, выраженная через кватернион, которые можно использовать для описания объекта в XML-файле. Если не хватает угла поворота по одной из осей, то добавьте его.

Узел **Material** содержит атрибут **Type**, который указывает на способ наложения материала на объект. В зависимости от способа наложения текстуры на объект **Type** может быть описан 3 способами, см. рис. 1.22:

- **smtSubMaterial** – используется в случае, если объект состоит из нескольких объектов и используется несколько материалов. Описание этих материалов находится в файле с расширением ***.material**. Каждый материал необходимо описывать в подузле с именем **SubMaterial**. В атрибуте **Name** узла **SubMaterial** указывается название материала. Атрибут **Index** узла **SubMaterial** – индекс, который указывает, какому из подобъектов назначается материал.
- **smtMaterial** – используется в случае, когда на объект накладывается один материал. В этом случае необходимо в атрибуте **Name** указать имя материала из файла с расширением ***.material**.
- **smtBitmap** – используется в случае, когда на объект накладывается текстура или текстурная карта без использования материала. В этом случае необходимо в атрибуте **TextureFileName** указать полное имя файла текстуры с его расширением.

```
<?xml version="1.0" encoding="utf-8"?>
<Base FileName="SampleUserBase.zip">
  <Group ID="2000000" Name="User Group">
    <Object Name="skoda_yeti" Description="Skoda Yeti" ID="2000001" Type="otMesh">
      <MeshFileName Value="Skoda_Yeti/skoda_yeti_2000001.mesh"/>
      <Position X="0" Y="0" Z="0.83"/>
      <Rotation X="0" Y="0.707" Z="0.707" W="0"/>
      <Scale Value="0.027"/>
      <Material Type="smtSubMaterial">
        <SubMaterial Name="Material_32_2000001" Index="0"/>
        <SubMaterial Name="Material_31_2000001" Index="1"/>
      </Material>
    </Object>
    <Object Name="Maple" Description="Maple" ID="2000002" Type="otMesh">
      <MeshFileName Value="Maple/maple_2000002.mesh"/>
      <Position X="0" Y="0" Z="0"/>
      <Rotation X="0.707106781186547" Y="0" Z="0" W="0.707106781186548"/>
      <Scale Value="0.4"/>
      <Material Type="smtMaterial" Name="maple_1_2000002"/>
    </Object>
    <Object Name="Bus_Stop_Sign" Description="Bus Stop Sign" ID="2000003" Type="otMesh">
      <MeshFileName Value="Bus_Stop_Sign/Bus_Stop_Sign_2000003.mesh"/>
      <Position X="0" Y="-0.215" Z="2"/>
      <Rotation X="1" Y="0" Z="0" W="0"/>
      <Scale Value="17"/>
      <Material Type="smtBitmap" TextureFileName="Bus_Stop_Sign/Bus_Stop_Sign_2000003.png"/>
    </Object>
    <Object Name="Grass" Description="Grass" ID="2000004" Type="otUnderlay">
      <Material Type="smtMaterial" Name="Grass_2000004"/>
    </Object>
  </Group>
</Base>
```

Рис. 1.22. Структура XML-файла

1.8.2.3. Связь ZIP-архивов и XML-файлов

Здесь рассмотрим несколько замечаний по общей структуре данных и порядке их загрузки. При старте **UM Simulation** сканирует все XML-файлы в каталоге *{Данные UM}\Scene data\User object* и загружает в окружение *Ogre3D* все модели из ZIP-архивов,

указанных в атрибуте **<Base/Filename>** просканированных XML-файлов, см. рис. 1.22. В одном XML-файле может быть только один узел **<Base>**. Таким образом, каждый XML-файл описывает содержимое только одного ZIP-архива. Имена XML-файлов могут отличаться от имён ZIP-архивов, которые они описывают. Вместе с тем для простоты поддержки пользовательской базы объектов рекомендуется, чтобы каждому ZIP-архиву соответствовал одноимённый XML-файл.

Обратите внимание, что в окружение *Ogre3D* загружаются только те ZIP-архивы, на которые есть ссылка из XML-файла. Если в вашем каталоге есть ZIP-архив, на который не ссылается ни один XML-файл, то данные из такого архива не загружаются и 3D объекты из него не будут доступны для использования в сценах.

1.9. Загрузка сцен в UM Simulation

Вся работа со сценами сосредоточена в программе **UM Simulation**. Для любой модели вы можете выбрать любую существующую сцену или разработать новую.

Как было сказано выше, ссылка на последнюю загруженную в модель сцену сохраняется в конфигурационном файле модели *.icf в структуре "with environment", поле "scenefilename". При загрузке модели поиск указанного в конфигурации файла сцены *.scene выполняется сначала по прямому пути, потом в каталоге модели и затем уже в каталоге {Данные УМ}\Scene data\Scenes. Учитывайте эту особенность работы программы при копировании ваших моделей и переносе их с компьютера на компьютер.

Мастер загрузки сцен, см. рис. 1.23, доступен в программе **UM Simulation** через пункт меню **Инструменты | Сцены**. Для загрузки файла сцены выберите нужный файл в поле **Путь к файлу активной сцены** или, если нужная сцена есть в списке, выберите пункт контекстного меню **Сделать сцену активной**. Используйте кнопку **Выгрузить активную сцену**, чтобы удалить элементы активной сцены из анимационного окна. Используйте кнопку **Обновить сцену**, чтобы заново загрузить активную сцену, если вы изменили её вручную или с помощью редактора сцен.

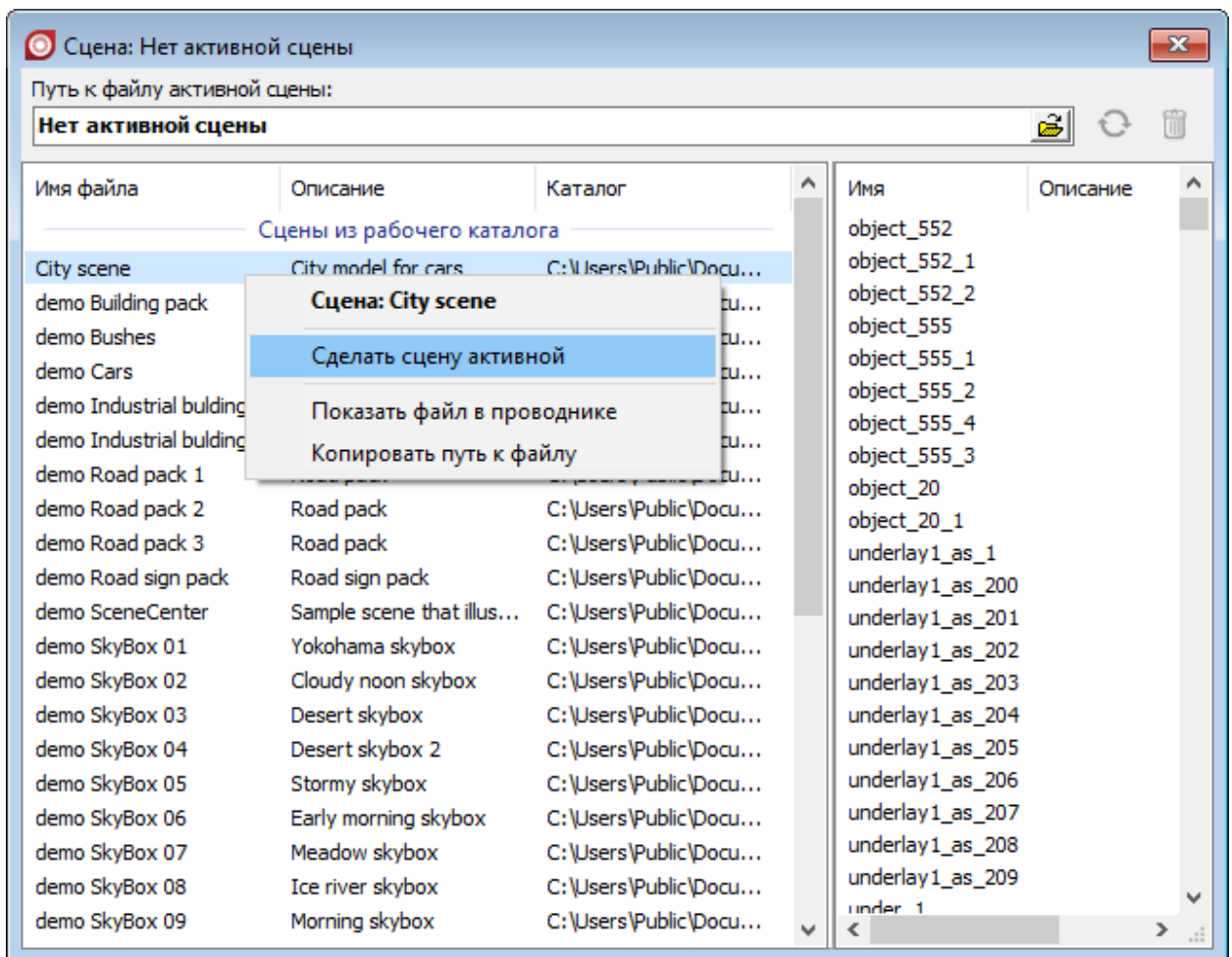


Рис. 1.23. Мастер загрузки сцен