

# Разработка модели данных в Erwin Data Modeler

Тема

# Основные особенности IDEF1X/ERWin

2

1. Поддерживается прямое (создание БД на основе модели) и обратное (генерация модели по имеющейся базе данных) проектирование для 20 типов СУБД.
2. Увеличивает производительность труда благодаря удобному интерфейсу и автоматизации рутинных процедур.
3. Поддерживает методологию структурного моделирования SADT и следующие нотации: IDEF1X.
4. Позволяет повторно использовать компоненты созданных ранее моделей, а также использовать наработки других разработчиков.

# Основные особенности IDEF1X/ERWin

3

- 5. Возможна совместная работа группы проектировщиков с одними и теми же моделями (с помощью AllFusion Model Manager 4.1).
- 6. Позволяет переносить структуру БД из одной СУБД в другую.
- 7. Позволяет документировать структуру БД.
- 8. Продукт можно использовать на всех стадиях жизненного цикла БД: проектировании, разработке, тестировании и поддержке.

# Уровни представления модели

## 4

- логический - это абстрактный взгляд на данные, на нем данные представляются так, как выглядят в реальном мире, и могут называться так, как они называются в реальном мире.
- физический - зависит от конкретной СУБД. В физической модели содержится информация о всех объектах БД. Одной и той же логической модели могут соответствовать несколько разных физических моделей.

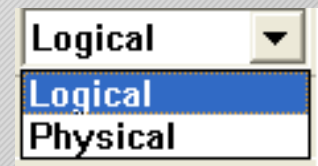
# Создание новой модели

5

- логическая
- физическая
- логико-физическая.

При проектировании базы данных целесообразнее выбирать логико-физическую модель.

Переключение между логической и физической моделью осуществляется на панели инструментов ERWin



# Уровни логической модели

6

В зависимости от представления информации о данных:

- диаграмма сущность-связь (Entity Relationship Diagram, ERD);
- модель данных, основанная на ключах (Key Based model, KB);
- полная атрибутивная модель (Fully Attributed model, FA).




# Этапы построения информационной модели

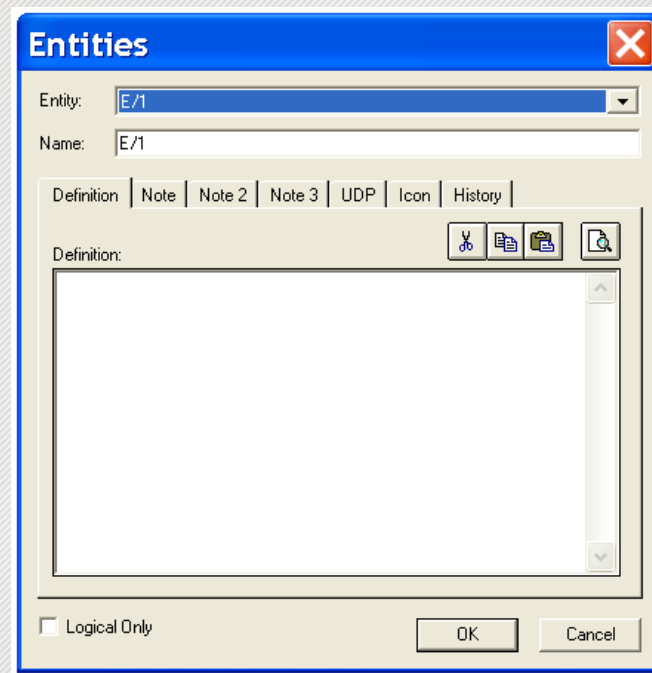
7

- определение сущностей;
- определение зависимостей между сущностями;
- задание первичных и альтернативных ключей;
- определение атрибутов сущностей;
- приведение модели к требуемому уровню нормальной формы;
- переход к физическому описанию модели: назначение соответствий имя сущности - имя таблицы, атрибут сущности - атрибут таблицы;
- задание триггеров, процедур и ограничений;
- генерация базы данных.

# Определение сущностей

8

-  Сущность на панели инструментов (ERWin Toolbox)
- Entity Editor:





# Определение сущностей

9

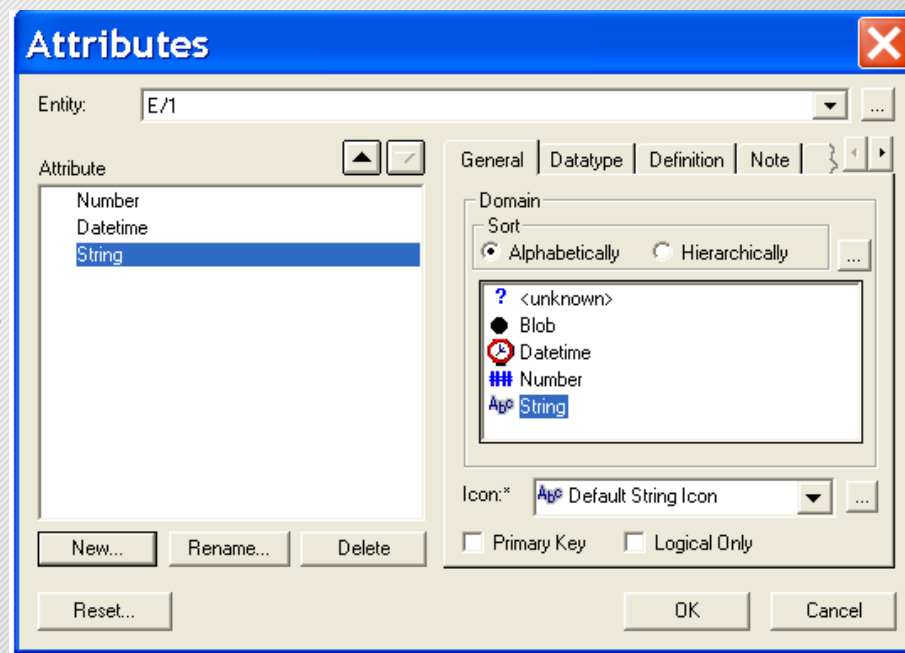
- Каждая сущность должна быть полностью определена с помощью текстового описания в закладке Definition.
- Вкладки Note, Note 2, Note 3, UDP (User Defined Properties - Свойства, определенные пользователем) служат для внесения дополнительных комментариев и определений к сущности.
- Вкладка Icon позволяет каждой сущности поставить в соответствие изображение, которое будет отображаться в режиме просмотра модели на уровне иконок.
- Вкладка History позволяет просмотреть историю всех изменений, связанных с сущностью и добавить комментарий к изменению в окне Comment.

# Определение атрибутов сущностей

10

## Attribute Editor

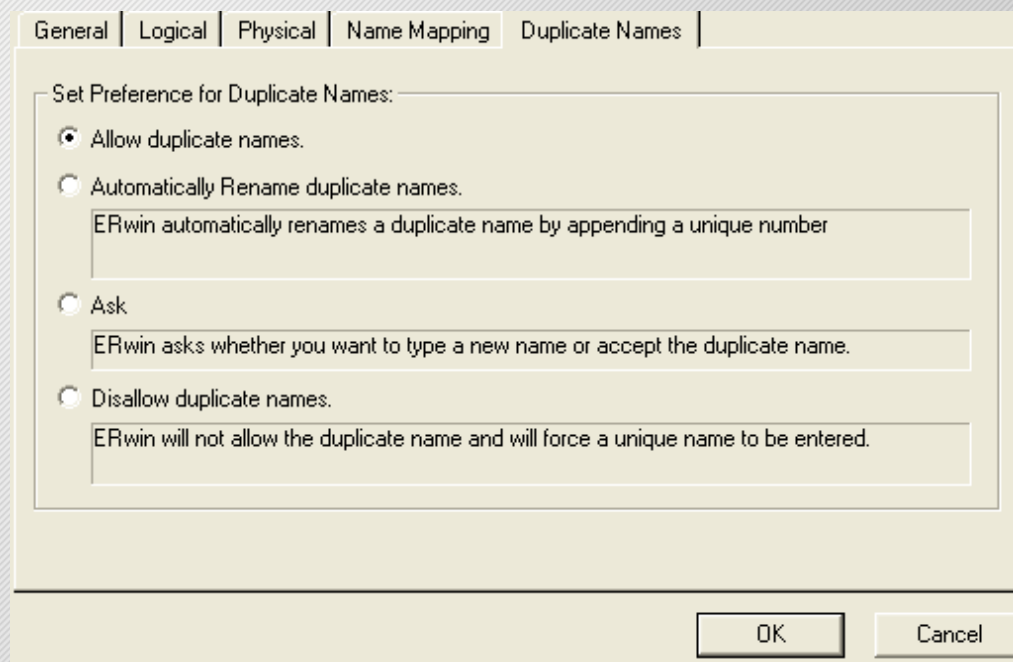
- Каждый атрибут хранит информацию об определенном свойстве сущности, а каждый экземпляр сущности должен быть уникальным.
- Атрибуты должны именоваться в единственном числе и иметь четкое смысловое значение



# Определение атрибутов сущностей

11

- Согласно синтаксису IDEF1X имя атрибута должно быть уникально в рамках модели (а не только в рамках сущности!).
- Меню Option/Unique Name: установить предпочтение



# Определение атрибутов сущностей: имя и домен

12

## General:

- имя атрибута
- имя соответствующей ему в физической модели колонки и домен.

**Домен** определяется как совокупность значений, из которых берутся значения атрибутов.

Каждый атрибут может быть определен только на одном домене, но на каждом домене может быть определено множество атрибутов.

В понятие домена входит не только тип данных, но и область значений данных.

# Определение атрибутов сущностей: ключи

13

- Сущность может иметь несколько возможных (потенциальных) ключей.
- Один потенциальный ключ объявляется первичным, а остальные могут быть объявлены альтернативными ключами.
- **Альтернативный ключ (Alternative Key)** - это потенциальный ключ, не ставший первичным.
- Для атрибутов первичного ключа необходимо установить флажок **Primary Key**.



# Определение атрибутов сущностей: ключи

14

- Сущность может иметь несколько возможных (потенциальных) ключей.
- Один потенциальный ключ объявляется первичным, а остальные могут быть объявлены альтернативными ключами.
- **Альтернативный ключ (Alternative Key)** - это потенциальный ключ, не ставший первичным.
- Для атрибутов первичного ключа необходимо установить флажок **Primary Key**.



# Определение атрибутов сущностей: индексы

15

- ERwin по умолчанию на физическом уровне генерирует уникальные индексы по первичному и альтернативным ключам.
- Индексы используются для поиска данных
- Индекс содержит отсортированную по колонке или нескольким колонкам информацию и указывает на строки, в которых хранится конкретное значение колонки
- ERwin позволяет также строить неуникальные индексы по атрибутам, называемым инверсными входами (Inversion Entries)
- Альтернативные ключи и инверсные входы создаются в диалоге Key Groups, который появляется после выбора соответствующего пункта контекстного меню

# Определение атрибутов сущностей

16

- Вкладка **Datatype** позволяет с помощью выпадающего списка на странице уточнить тип данных для выделенного атрибута
- Вкладка **Constraint** позволяет задать ограничения для выделенного атрибута.
- Вкладка **Definition** позволяет записать определения отдельных атрибутов.
- Вкладка **Note** позволяет добавлять замечания об одном или нескольких атрибутах сущности, которые не вошли в определения.
- Вкладка **UDP** предназначена для задания значений свойств, определяемых пользователем.

# Определение атрибутов сущностей

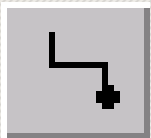
17

- ERwin позволяет переносить атрибуты внутри и между сущностями.
- При переносе атрибутов внутри и между сущностями можно воспользоваться техникой drag&drop.

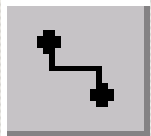
# Создание связей

18

- Тип связи



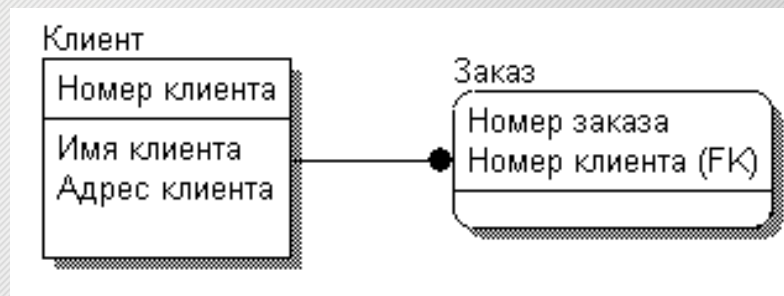
идентифицирующая



многие ко многим возможна только на уровне логической модели данных

неидентифицирующая

- При установлении идентифицирующей связи атрибуты первичного ключа родительской сущности автоматически переносятся в состав первичного ключа дочерней сущности - миграция атрибутов.
- В дочерней сущности новые атрибуты помечаются как внешний ключ - (FK).

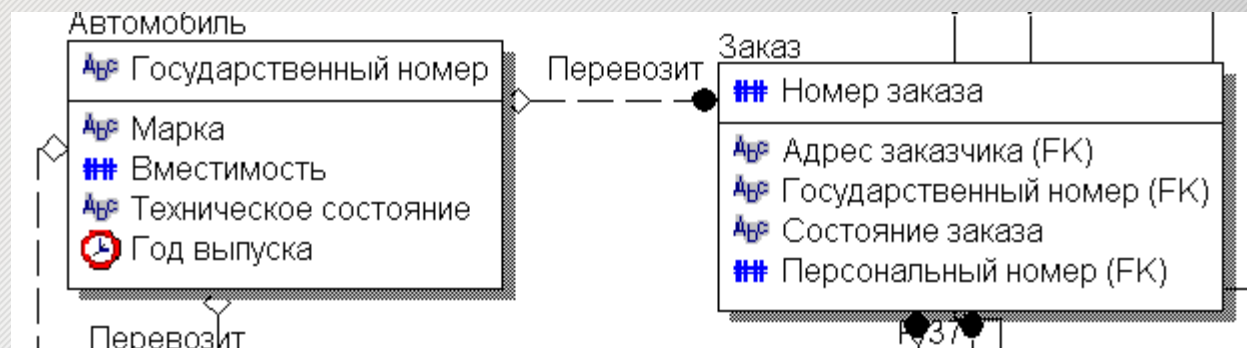




# Создание связей

20

- При установлении неидентифицирующей связи атрибуты первичного ключа родительской сущности мигрируют в состав неключевых компонентов дочерней сущности.
- Неидентифицирующая связь служит для связывания независимых сущностей.





# Создание связей : General

21

## General:

МОЩНОСТЬ, ИМЯ И ТИП СВЯЗИ.

*Имя связи (Verb Phrase)* - фраза, характеризующая отношение между родительской и дочерней сущностями.

Для неидентифицирующей связи указывается обязательность (Nulls или No Nulls).

The screenshot shows the 'Relationships' dialog box with the 'General' tab selected. The 'Relationship' dropdown is set to 'Клиент Заказ'. The 'Verb Phrase' section has two empty text boxes for 'Parent-to-Child' and 'Child-to-Parent'. The 'Relationship Cardinality' section shows a summary of 'One-to-Zero-One-or-More'. Under 'Cardinality', the 'Zero, One or More' radio button is selected. Under 'Relationship Type', the 'Non-Identifying' radio button is selected. The 'Nulls' section has the 'No Nulls' radio button selected. The 'Logical Only' checkbox is unchecked. The 'New...' and 'Delete' buttons are visible at the top right, and 'OK' and 'Cancel' buttons are at the bottom right.

# Создание связей: RI Action

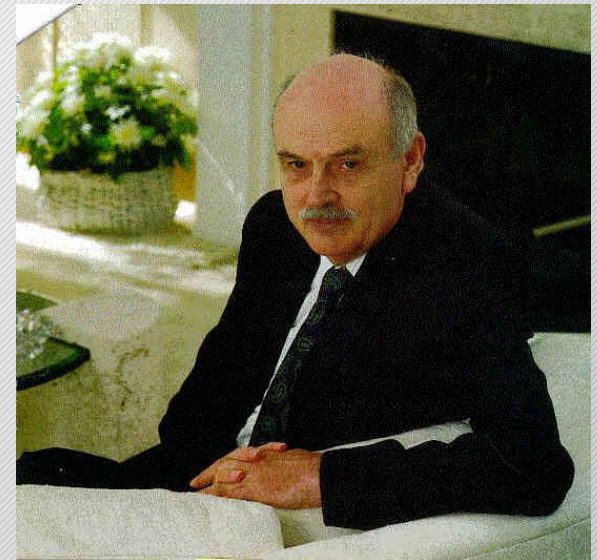
22

- **RI (Referential Integrity) Action** (Правила ссылочной целостности): для каждого значения внешнего ключа должно существовать соответствующее значение первичного ключа в родительском отношении.  
Room (Id, Owner, Status) (101, 12, 0)  
Owner (Id, Login, Password, Name, Last name) (12, abcd, 1234,...)

# Реляционные базы данных

35

- Реляционная модель данных предложена сотрудником фирмы IBM Эдгаром Коддом (Edgar Codd) в начале 70-х годов.



Любое представление данных сводится к совокупности двумерных таблиц особого вида, известного в математике как отношение (relation).

- Сущности представляются в виде таблицы
- Каждому атрибуту соответствует столбец таблицы
- Конкретному экземпляру объекта сопоставляется строка таблицы (запись или кортеж)

# Создание связей: RI Action

25

- Ссылочная целостность может нарушиться в результате операций с экземплярами сущностей (кортежами или записями в таблице):
  - вставка,
  - обновление,
  - удаление.



# Создание связей: RI Action

26

## *Родительская сущность*

- **Вставка** записи в таблицу не нарушает ссылочной целостности, так как допустимо существование кортежа (экземпляра сущности) в родительской таблице (сущности), на который нет ссылок из дочерних таблиц (сущностей)
- **Обновление** записи (изменение атрибутов экземпляра сущности) может привести к нарушению ссылочной целостности при изменении значения первичного ключа. Если есть записи в дочерней таблице (экземпляры) дочерней сущности, ссылающиеся на обновляемую запись в родительской таблице, то значения их внешних ключей станут некорректными.



# Создание связей: RI Action

27

## *Родительская сущность*

- **Удаление** записи родительской таблицы (экземпляра сущности) может привести к нарушению ссылочной целостности, так как удаляется значение первичного ключа. Если есть записи дочерней таблицы (экземпляры сущности), ссылающиеся на удаляемую запись родительской таблицы отношения, то значения их внешних ключей станут некорректными.

# Создание связей: RI Action

28

## *Дочерняя сущность*

- **Вставка** записи в таблицу может привести к нарушению ссылочной целостности, если вставляемое значение внешнего ключа некорректно.
- **Обновление** записи в дочерней таблице может привести к нарушению ссылочной целостности при некорректном изменении значения внешнего ключа.
- При удалении записи дочерней таблицы ссылочная целостность не нарушается.

# Создание связей: основные стратегии

29

- 1) **RESTRICT** (ОГРАНИЧИТЬ) - не разрешать выполнение операции, приводящей к нарушению ссылочной целостности.
- 2) **CASCADE** (КАСКАД) - разрешить выполнение требуемой операции, но внести при этом необходимые поправки так, чтобы не допустить нарушения ссылочной целостности и сохранить все имеющиеся связи.

Изменение начинается в родительской таблице и каскадно выполняется в дочерней. Так как дочерняя таблица может быть родительским для некоторой третьей таблицы, то может потребоваться выполнение каскадной стратегии и для этой связи и т.д.

# Создание связей: дополнительные стратегии

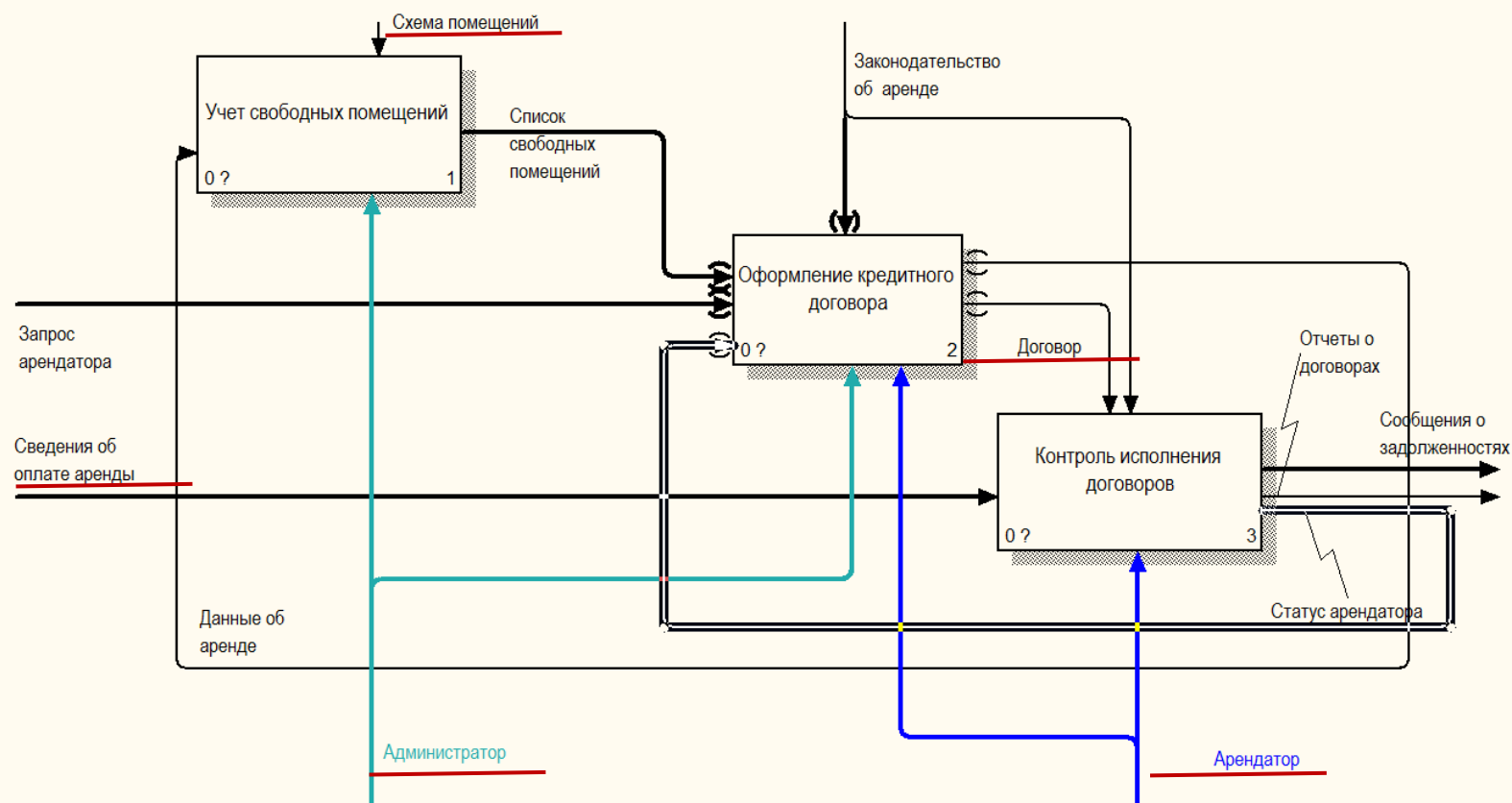
30

- 1) **NONE** (НИКАКОЙ) - никаких операций по поддержке ссылочной целостности не выполняется.
- 2) **SET NULL** (УСТАНОВИТЬ В NULL) - разрешить выполнение требуемой операции, но все возникающие некорректные значения внешних ключей заменять на неопределенные значения (null-значения). При этом кортежи дочернего отношения теряют всякую связь с кортежами родительского отношения.
- 3) **SET DEFAULT** (УСТАНОВИТЬ ПО УМОЛЧАНИЮ) - разрешить выполнение требуемой операции, но все возникающие некорректные значения внешних ключей изменять на некоторое значение, принятое по умолчанию. При этом должна существовать запись в родительской таблице, первичный ключ которой принят как значение по умолчанию для внешних ключей, которую нельзя удалять и изменять значение ключа.

# Пример моделирования

31

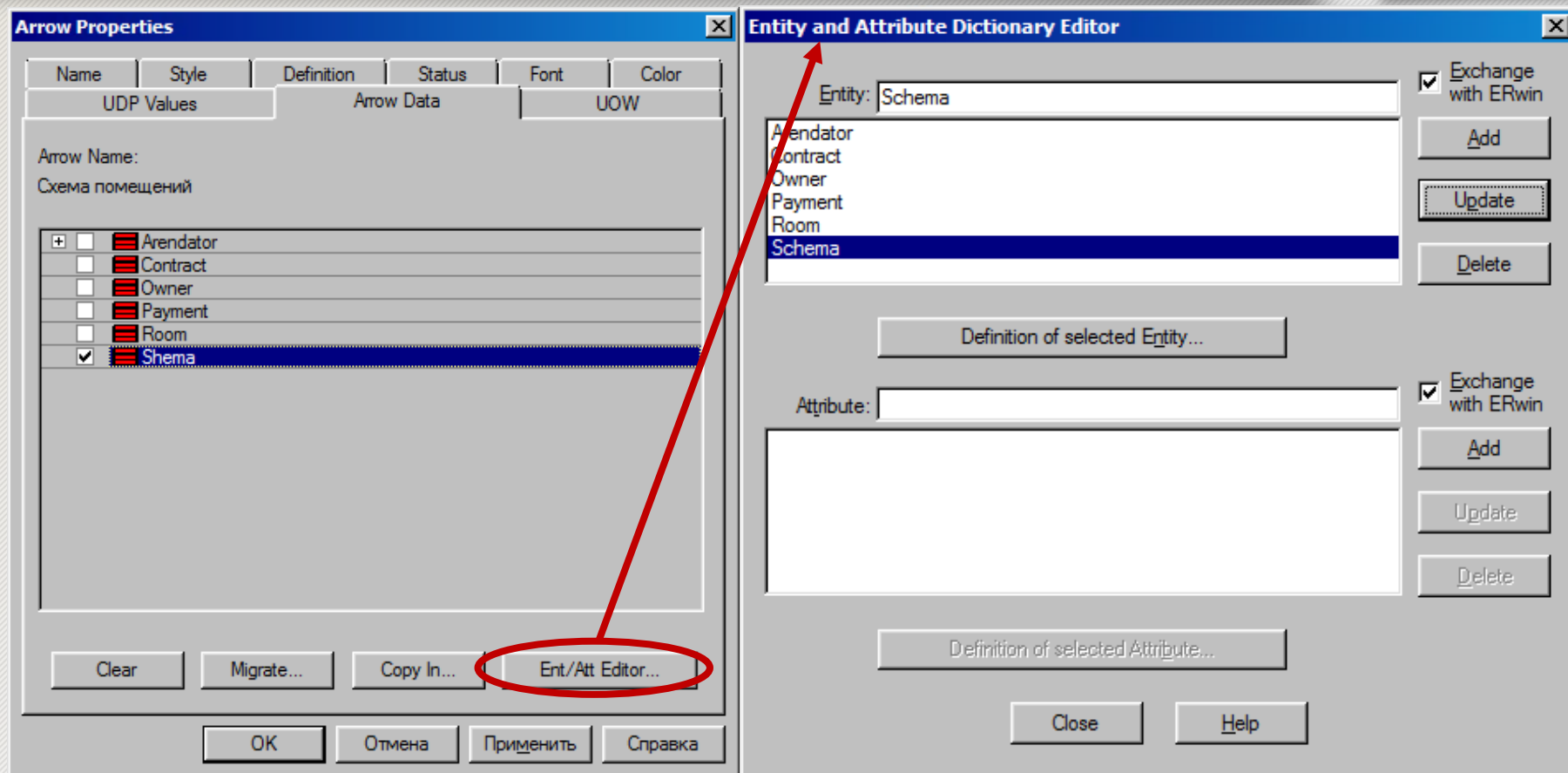
USED AT:	AUTHOR:	DATE: 14.09.2017	WORKING	READER	DATE	CONTEXT:
	PROJECT: Άόάίαά μάάάίάέ	REV: 30.03.2020	DRAFT			
			RECOMMENDED			
	NOTES: 1 2 3 4 5 6 7 8 9 10		PUBLICATION			A-0





# Свойства стрелок: данные

32





# Редактор словарей

33

Entity and Attribute Dictionary Editor

Entity: Contract

☒ Exchange with ERwin

Add

Update

Delete

Definition of selected Entity...

Attribute: CosrPerMonth

☒ Exchange with ERwin

Add

Update

Delete

Definition of selected Attribute...

Close Help

Определение атрибутов  
для выделенных  
сущностей

# Словари сущностей и атрибутов

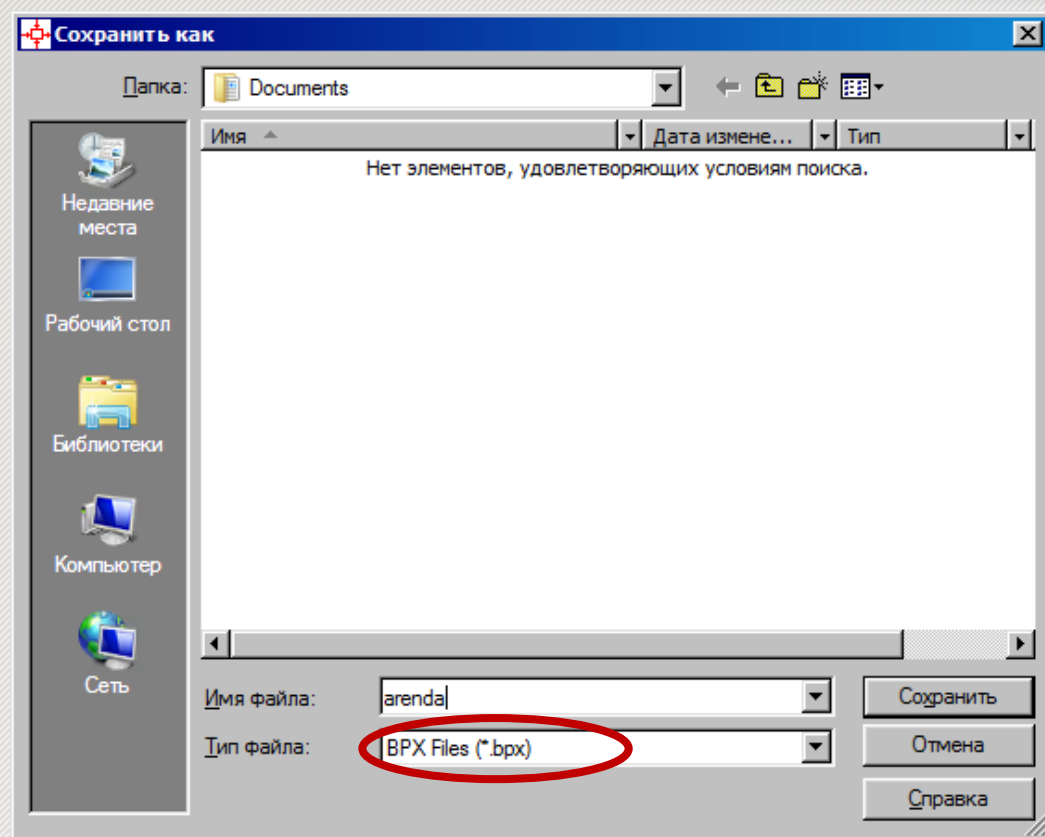
34

- Dictionary - Entity
- Dictionary - Attribute

Attribute Dictionary			
Name	Definition	Exchange with ERwin	Entity
Arendator		<input checked="" type="checkbox"/> Exchange with ERwin	Contract
CosrPerMonth		<input checked="" type="checkbox"/> Exchange with ERwin	Contract
DateEnd		<input checked="" type="checkbox"/> Exchange with ERwin	Contract
DateStart		<input checked="" type="checkbox"/> Exchange with ERwin	Contract
Floor		<input checked="" type="checkbox"/> Exchange with ERwin	Room
ID		<input checked="" type="checkbox"/> Exchange with ERwin	Arendator
ID		<input checked="" type="checkbox"/> Exchange with ERwin	Room
ID		<input checked="" type="checkbox"/> Exchange with ERwin	Contract
LastName		<input checked="" type="checkbox"/> Exchange with ERwin	Arendator
Login		<input checked="" type="checkbox"/> Exchange with ERwin	Arendator
Name		<input checked="" type="checkbox"/> Exchange with ERwin	Arendator
Owner		<input checked="" type="checkbox"/> Exchange with ERwin	Room
Owner		<input checked="" type="checkbox"/> Exchange with ERwin	Contract
Password		<input checked="" type="checkbox"/> Exchange with ERwin	Arendator
Room		<input checked="" type="checkbox"/> Exchange with ERwin	Contract
Square		<input checked="" type="checkbox"/> Exchange with ERwin	Room
Status		<input checked="" type="checkbox"/> Exchange with ERwin	Arendator
		<input checked="" type="checkbox"/> Exchange with ERwin	Room

# Экспорт данных в Erwin Data Modeler

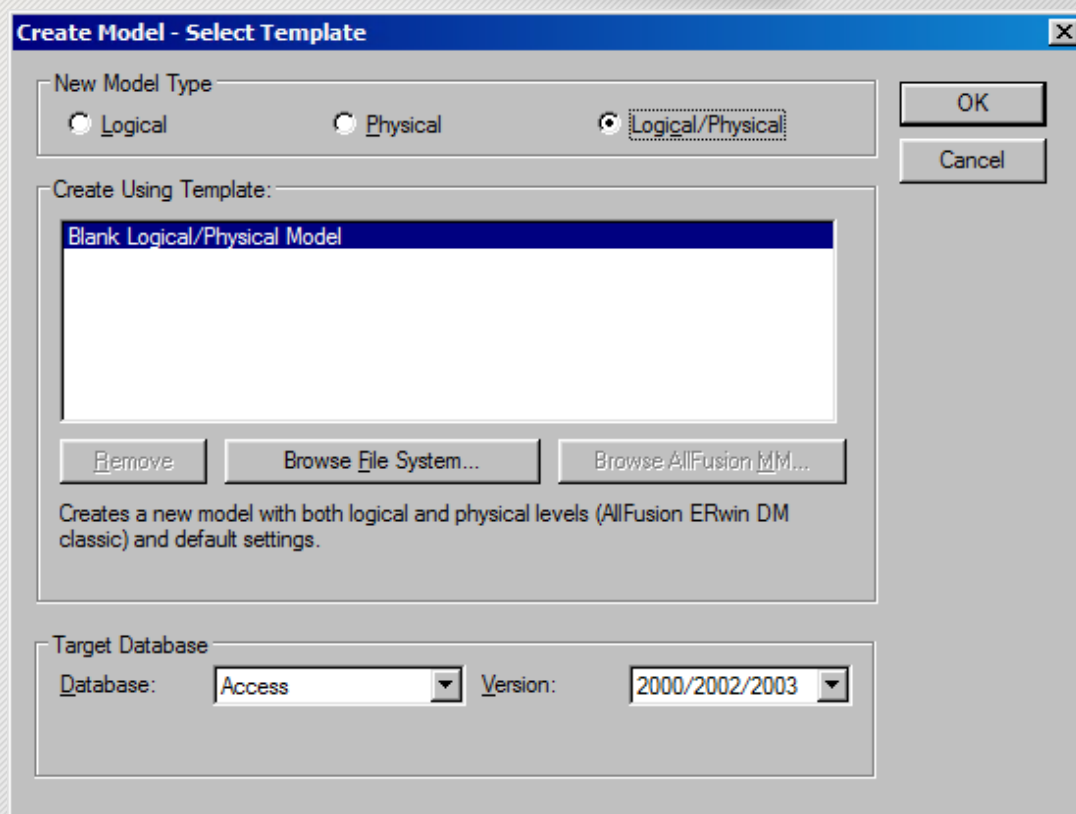
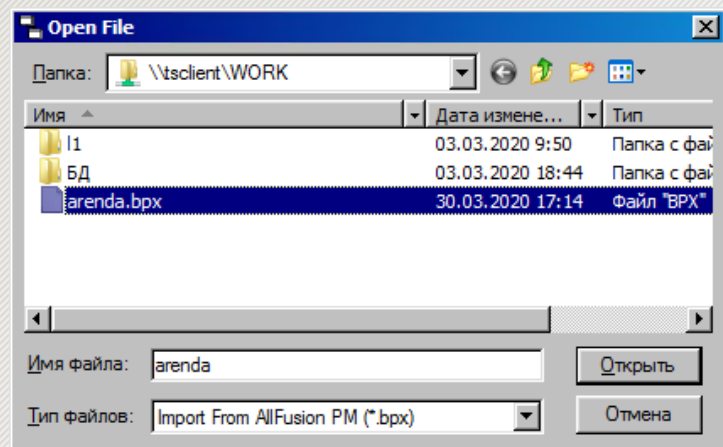
35



# Создание проекта и импорт данных

36

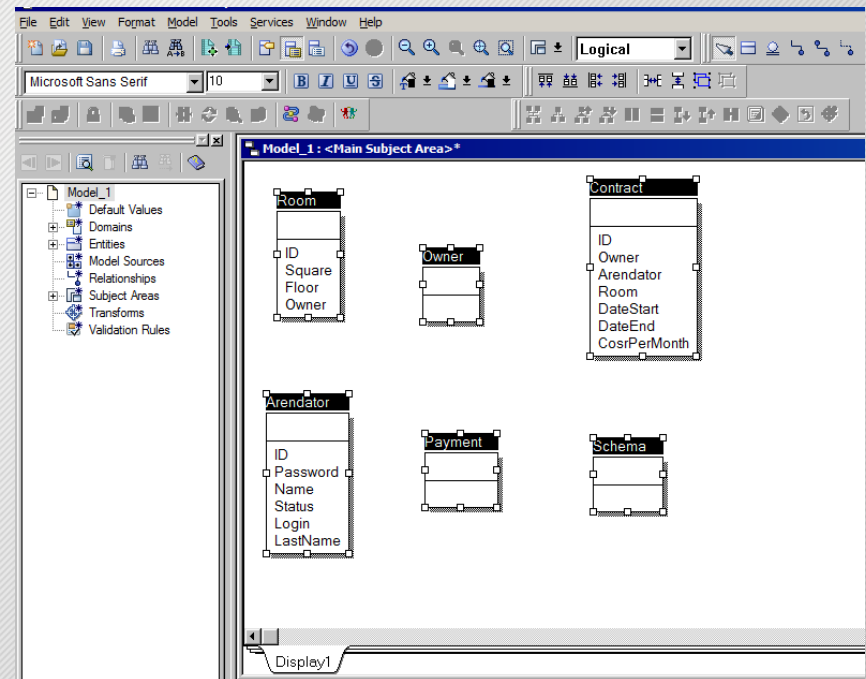
- New model - Logical/Physical
- Import from Allfusion Process Modeler



# Импорт сущностей

37

- **Contract** (Id, Room, Arendator)
- **Room** (Id, Owner, Status)
- **Arendator** (Id, Login, Password, Name, Last name)
- **Owner** (Id, Login, Password, Name, Last name)
- **Payment** (Id, Contract, Arendator)





# Создание связей: general

38

## *Owner-to-Room*

- Тип связи: **неидентифицирующая**
- Направленность: **Owner (parent) to Room (child)**
- Имя связи (Verb Phrase): **parent-to-child → owns**  
**child-to-owner → belong to**
- Мощность (Cardinally): **zero, one or more**  
(может владеть ни одним, одним или несколькими)
- Обязательность со стороны дочерней сущности (Nulls): **nulls** (есть помещения без владельца)

# Создание связей: Rolename

39

- **Definition** – полное определение связи
- **Rolename** – имя роли атрибута внешнего ключа (обязательно для рекурсивных связей)
- **Rolename** (Имя роли, функциональное имя) - это синоним атрибута внешнего ключа, который показывает, какую роль играет атрибут в дочерней сущности.

The screenshot shows a 'Relationships' dialog box with the 'Rolename' tab selected. The 'Relationship' dropdown is set to 'R/2 (Owner to Room)'. The 'Name' field contains 'R/2'. There are 'New...' and 'Delete' buttons. The 'Rolename Info' section contains a 'Migrated Key' dropdown set to 'XPKOwner', a 'Migrated Attribute' field, and a 'Rolename' field. At the bottom are 'OK' and 'Cancel' buttons.

# Создание связей: ссылочная целостность

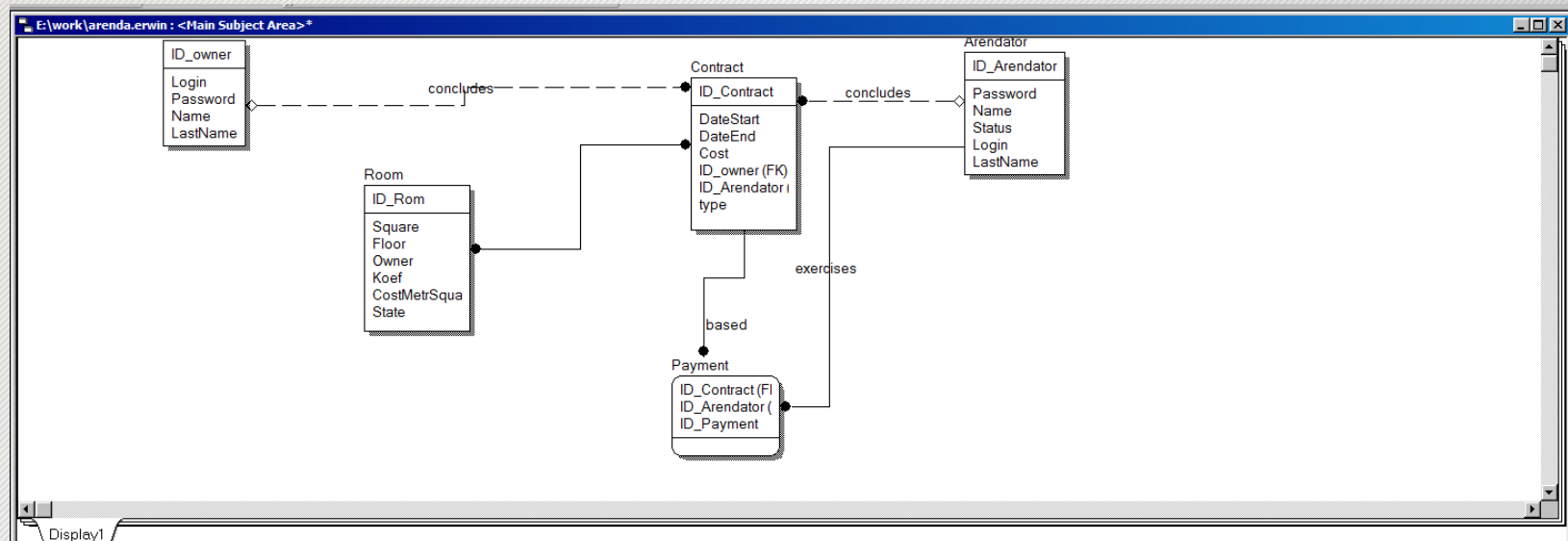
41

- **RI (Referential Integrity) Action** (Правила ссылочной целостности)
- **Child Delete** (Удаление экземпляра дочерней сущности)-**None** (проверка не производится)
- **Child Insert** (Вставка экземпляра дочерней сущности)-

The screenshot shows a 'Relationships' dialog box with the following details:

- Relationship:** R/2 (Owner to Room)
- Name:** R/2
- Buttons:** New..., Delete
- Tabs:** General, Definition, Rolename, RI Actions (selected), UDP
- RI Actions Section:**
  - Child Delete:** NONE
  - Parent Delete:** SET NULL
  - Child Insert:** SET NULL
  - Parent Insert:** NONE
  - Child Update:** SET NULL
  - Parent Update:** A dropdown menu is open, showing options: RESTRICT, CASCADE, SET NULL, SET DEFAULT, NO ACTION, and NONE (highlighted).
- Buttons at the bottom:** OK, Cancel

- **Owner** and **Arendator** подтипы супер типа **USER**



# Преобразование связи

43

Supertype:

Subtypes:

Type: ☒ Complete ☐ Incomplete

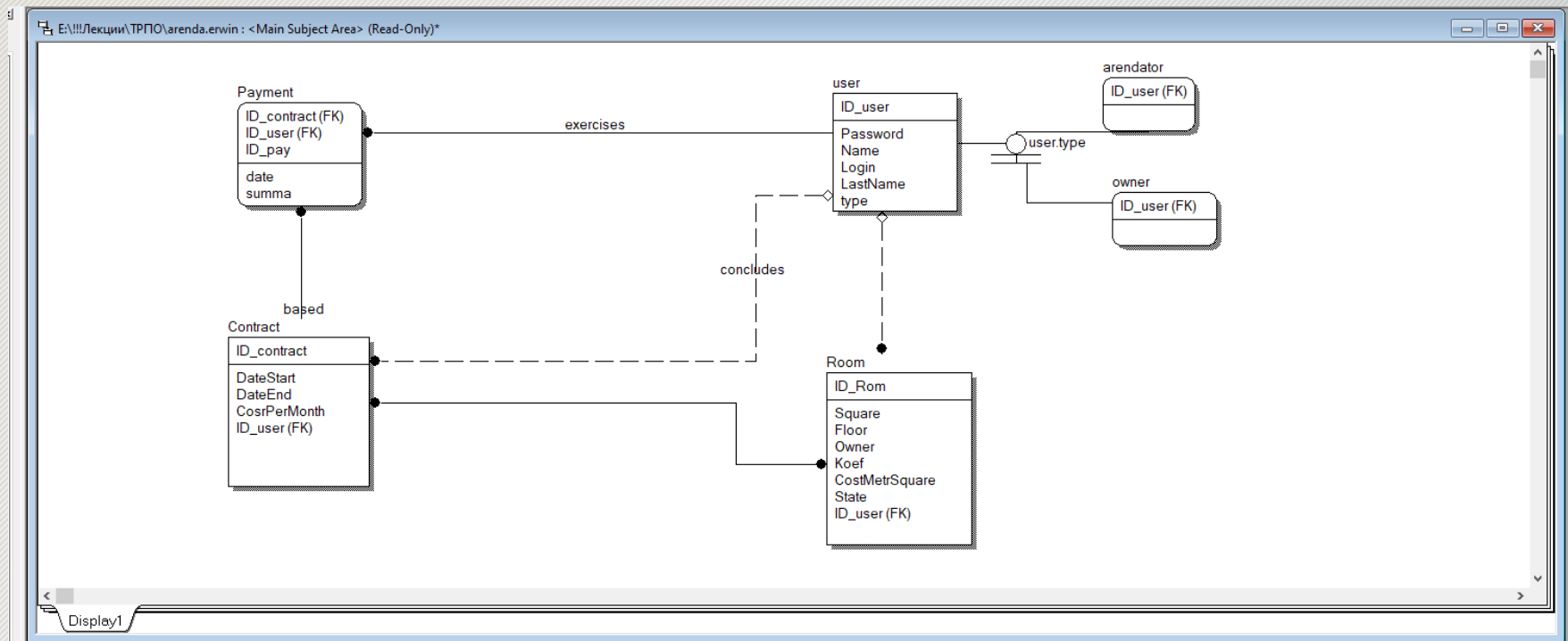
Discriminator:

- ☐ user
  - ☐ ID\_user
  - ☐ Password
  - ☐ Name
  - ☐ Login
  - ☐ LastName
  - ☒ type



# Типизирующая связь

44



# Сущности и атрибуты логической модели

45

Attributes

Entity: user

Attribute

- ID\_user
- Password
- Name
- Login
- LastName
- type

General | Datatype | Constraint | Definit... < >

Domain

Sort

☒ Alphabetically ☐ Hierarchically

- <default>
- Blob
- Datetime
- Number
- String

Icon: \* Default String Icon

☐ Primary Key ☒ Logical Only

OK Cancel

Entities

Entity: arendator

Name: arendator

Definition | Volumetrics | Note | Note 2 | Note 3 | UDP | Icon | History

Note:

☒ Logical Only

OK Cancel

# Таблица физической модели

46

Access Tables

Table: role

Name: role Owner:

Comment | Volumetrics | UDP | History | Validation

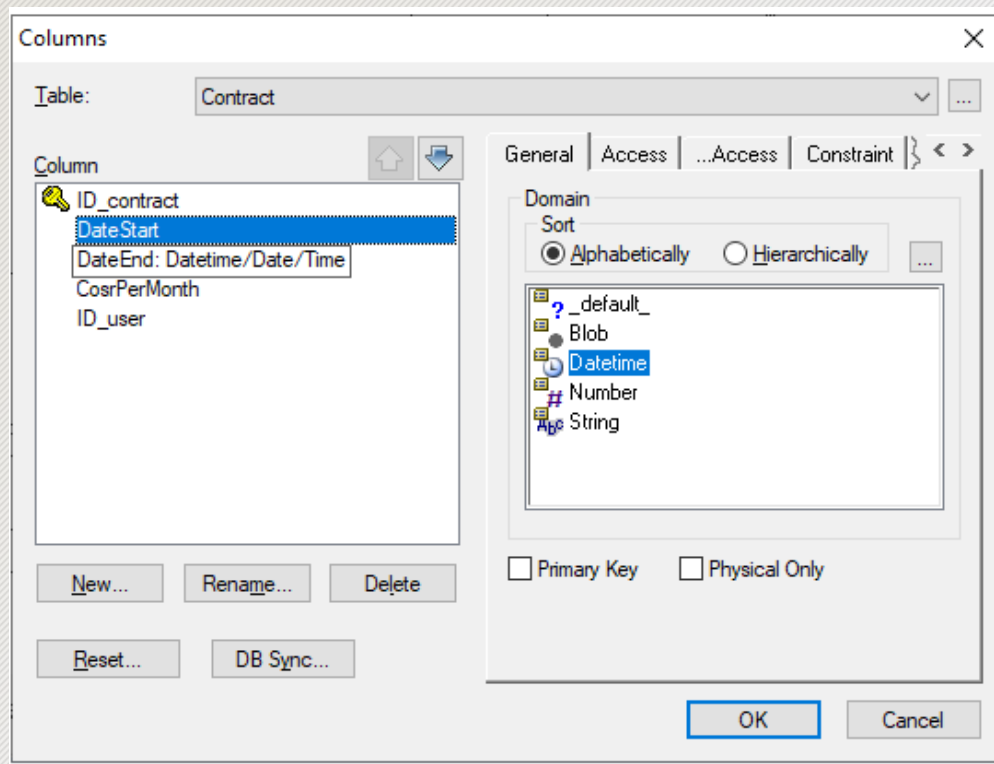
Comment:\*

☐ Update Entity Definition To Match

☒ Physical Only ☒ Generate DB Sync... OK Cancel

# Столбцы физической модели

47



# Физическая модель

48

