

В. Дихтяр

# ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

(для бакалавров)

Раздел 1. *Системы управления базами данных и базами знаний*

Тема 1-3. *Реляционный подход к построению информационно-логической модели (ИЛМ)*

---

---

*Скорость роста качества продукта и скорость совершенствования бизнес-процессов будут намного выше, а при достаточно большом значении этих показателей произойдет изменение характера самого бизнеса.*

*Б. Гейтс*

# ВВЕДЕНИЕ

---

- 1980 → первые коммерческие версии СУРБД
- 2000 → в большей части  $\Phi$  (во всех сферах  $A$ ): хранение, ведение и анализ  $\check{D}$

Идея:  $\check{D}$  → в  $T$  (таблица)

- столбцы  $T^j = \underline{\text{поля (атрибуты, домены)}}$ : один тип  $\check{D}$
- строки  $T_i = \underline{\text{записи}}$ : значения, соответствующие  $T^j$

# ТАБЛИЦЫ

---

$T \equiv$  двухмерный массив ( $t_{ij}$ ):

- $\forall t_{ij}$  – один элемент  $\check{D}$
- все  $T^j$  однородные
- $\forall T^j$  имеет уникальное имя
- одинаковых  $T_i$  нет
- порядок следования  $T_i$  и  $T^j$  – произвольный

# ИНФОРМАЦИОННЫЙ ОБЪЕКТ $\hat{o}_i$

---

- ≡ описание некоторой сущности ( $\hat{o}_{real}$ )
- реквизитный состав +  $\hat{S}(\hat{o}_i)$ 
  - ⇒ *класс* (тип) + *уникальное имя* (обозначение)
- $\hat{o}_i$  имеет множество реализаций – *экземпляров*
- экземпляр ≡ {конкретные значения реквизитов}, идентифицируется ключом (*простой* – один реквизит, *составной* – несколько)
- остальные реквизиты – описательные

# СТРУКТУРА ТАБЛИЦ $\hat{S}(T)$

$T$  - фундаментальный элемент БД  
(соответствует одной сущности)

- $\hat{S}(T) \equiv \{\text{связанные } T^j\}$ ; связь  $\equiv$  отношение  $T^j$
- значения  $T^j$  атомарные  
 $\neq \{\text{массивы или } \hat{S} \text{ значений}\}$
- типы хранимых значений:  
алфавитно-цифровые, цифровые, «дата», ..

$\hat{S}$  - структура;  $T$  - период долго/краткосрочный;  $\hat{o}$  - объект;

# ПРИМЕР 1

CUSTOMER			
CUST_NR	NAME	E-MAIL	
1	John Doe	<NULL>	
2	Mary Doe	m@aha.co	
...	...	...	

PLASE_ORDERS	
CUST_NR	ORDER_NR
1	367
1	490
...	...

ORDER	
ORDER_NR	ORDER_DATE
...	...
367	2001-01-17
...	...
490	2001-01-29
...	...

# КЛЮЧИ

---

- $T_V \equiv$  первичный (простой) ключ = один из  $T^j$
- ( $\forall$  комбинация  $\equiv$  составной ключ)

*подчеркиваются  $\Rightarrow$*

- *однозначно* идентифицировать  $\forall T_i$

(правило целостности)  $\Rightarrow$

- различимость всех  $T_i$
- СВЯЗЬ  $T1_i \rightarrow T2_k$



# ПРИМЕР 1 (ПРОДОЛЖЕНИЕ)

---



*T* customer:  $T_V = \underline{\text{CUST\_NR}}$ ,

*T* order:  $T_V = \underline{\text{ORDER\_NR}}$ ,

*T* place\_orders:  $T_V = \underline{\text{CUST\_NR} + \text{ORDER\_NR}}$

(комбинация )



# ОРГАНИЗАЦИЯ

---

- $T \supset \check{D}$  | организованы по  $T_i = \{ \text{значения } T^j \}$
- $\Rightarrow T_i$  определяет некоторую *сущность*  $\equiv \hat{O}$
- значений нет  $\Rightarrow$  «NULL»

# ВНЕШНИЙ КЛЮЧ $T_U$

- $T_U = T^j$  ( $\forall$  комбинация)  $\in T1 \Rightarrow$  ссылка на  $T_i \in T2$   
( $T1 =$  источник,  $T2 =$  целевая, родительская)  $\Rightarrow$   
 $T_U = \{ \text{значения } T_V - \text{первичных ключей } \in T2 \}$

*Правило* «ссылочной целостности»:  $T_U$  (или его части)

1. не имеет значения (= содержит *NULL*)  $\forall$
2. имеет значение (во всех  $T^j$ ) и оно обязательно должно содержаться в значениях  $T_V(T2)$

# $T_U$ ПО ОТНОШЕНИЮ К СТРОКАМ

- ограничение изменения или удаления: пока  $\exists$  ссылка по  $T_U$  на конкретную  $T_i(T2)$ , значение  $T_V(T2)$  не может быть изменено или удалено
- каскадное изменение или удаление: значение  $T_V(T2)$  изменяется  $\Rightarrow$  значения  $T_U(T1)$  соответствующих строк тоже изменяются;  $T_i(T2)$  удаляется  $\Rightarrow$  все связанные с ней по  $T_U$   $T_i(T1)$  тоже удаляются
- обнуление обновления или удаления:  $T_V(T2)$  изменяется  $\Rightarrow T_i$  удаляется +  $T_U$  соответствующих  $T_i(T1)$  теряют свои значения (*NULL*)

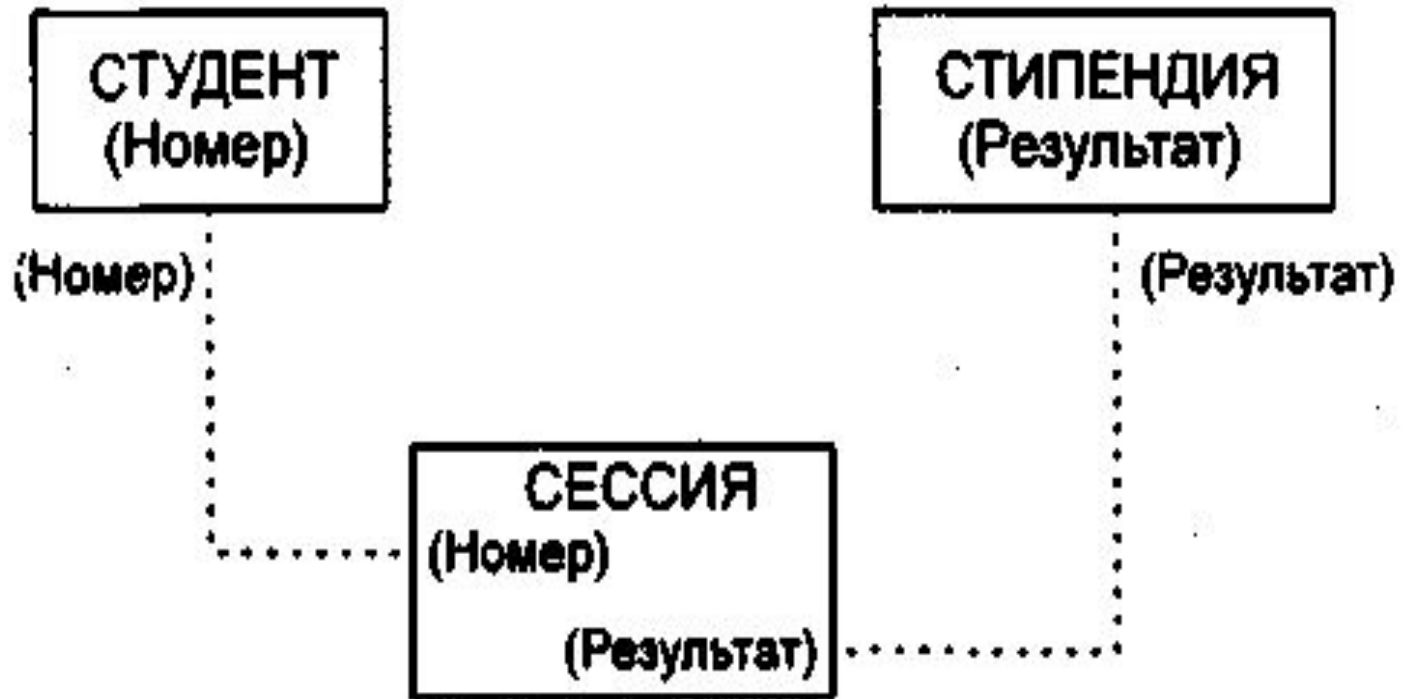
# ПРИМЕР 1 (ПРОДОЛЖЕНИЕ)

□  $T_{PLACE\_ORDERS} \Rightarrow$  первичный ключ  $T_V = (T1_U, T2_U)$  - комбинация двух внешних ключей

$T_V\_CUST\_NR \equiv T1_U \rightarrow T_V =$   
 $CUST\_NR \in T_{CUSTOMER}$

$T_V\_ORDER\_NR \equiv T2_U \rightarrow T_V =$   
 $ORDER\_NR \in T_{ORDERS}$  □

# ПРИМЕР 2: РЕЛЯЦИОННАЯ МОДЕЛЬ



## ПРИМЕР 2: ПОЯСНЕНИЕ

---

- $T1$  = СТУДЕНТ (*Номер*, Фамилия, Имя, Отчество, Пол, Дата рождения, Группа)
- $T2$  = СЕССИЯ (*Номер*, Оценка 1, Оценка 2, Оценка 3, Оценка 4, *Результат*)
- $T3$  = СТИПЕНДИЯ (*Результат*, Процент)

$T1 \leftrightarrow T2 : T_V = \text{Номер}$

$T2 \leftrightarrow T3 : T_U = \text{Результат}$

# НОРМАЛИЗАЦИЯ ОТНОШЕНИЙ

- ≡ формальный аппарат ограничений на формирование отношений (таблиц) ⇒
- устранить дублирование
  - непротиворечивость хранимых в *БД*
  - ↓ трудозатраты на ведение *БД*



# ФОРМЫ НОРМАЛИЗАЦИИ: ПЕРВАЯ НФ

определяет понятие **T**:

- фиксированное число столбцов
- все атрибуты простые (неделимые) =  
элементарные значения

*Требование 1НФ, по сути, повторяет свойство  
реляционных таблиц: каждый элемент таблицы  
— один элемент данных.*

□ **T** Студент = (Номер, Фамилия, Имя, Отчество,  
Группа, Дата) □

## ВТОРАЯ НФ

---

- $\equiv$  1 форма +  $\forall$  неключевой атрибут функционально полно ( $f$ -полно) зависит от составного ключа
- $f$ -зависимость: в экземпляре  $\hat{o}_i$  определенному значению ключевого реквизита соответствует **ТОЛЬКО ОДНО** значение описательного реквизита (атрибута)
  - $T^j$   $f$ -зависит от  $T_V$ :  $\forall$  значению  $T_V$  определено значение  $T^j \Rightarrow$  обозначение « $T_V - T^j$ » □

## ***f**-ПОЛНАЯ ЗАВИСИМОСТЬ*

- ≡  $\forall$  неключевой атрибут *f*-зависит от ключа (но ни от какой части составного ключа)
- **T** Студент = (Номер, Фамилия, Имя, Отчество, Дата, Группа): 1нф и 2 нф одновременно описательные реквизиты однозначно определены и *f*-зависят от  $T_V = \text{Номер}$  □
  
  - **T** Успеваемость = (Номер, Фамилия, Имя, Отчество, Дисциплина, оценка): 1нф, составной  $T_V = \text{Номер} + \text{Дисциплина}$  не находится во 2нф: Фамилия, Имя, Отчество не находятся в полной *f*-зависимости с  $T_V$  □

# ПРИМЕР (упр.)

Первичный ключ: CUST\_NR и ORDER\_NR

*f*- зависимости:

- (CUST\_NR, ORDER\_NR) (NAME, ORDER\_DATE): все столбцы, не являющиеся первичными ключами, зависят от первичного ключа
- CUST\_NR NAME: значения NAME зависят только от CUST\_NR
- ORDER\_NR ORDER\_DATE: ORDER\_DATE зависят только от ORDER\_NR

CUSTOMER_ORDERS			
NAME			
John Doe	1	367	2001-01-17
Mary Doe	2	480	2001-01-29
...	...	...	...

CUSTOMER_ORDERS	
CUST_NR	ORDER_NR
1	367
2	480
...	...

CUSTOMER	
NAME	CUST_NR
John Doe	1
Mary Doe	2
...	...

ORDERS	
ORDER_NR	ORDER_DATE
367	2001-01-17
480	2001-01-29
...	...

## ТРЕТЬЯ НФ

---

Транзитивная зависимость  $\equiv$  если один из 2-х описательных реквизитов зависит от ключа, а другой от 1-го описательного реквизита

3 нф  $\equiv$  2нф+  $\forall$  неключевой атрибут  
*нетранзитивно* зависит от первичного ключа

## ПРИМЕР 3 нф (упр)

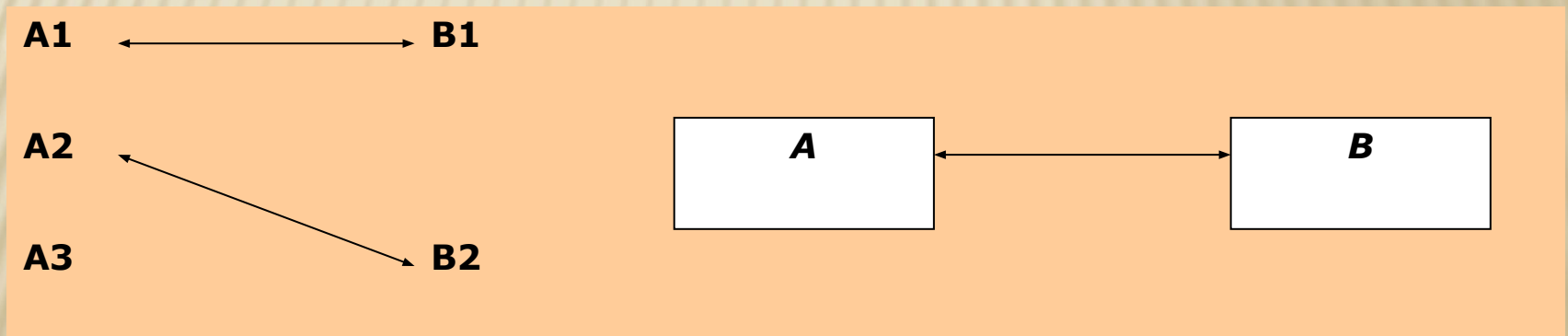
---

- *Транзитивная зависимость*
- **T** Студент: + Староста (определяется только номером группы)  $\Rightarrow$  фамилия старосты будет многократно повторяться в разных экземплярах  $\hat{O}_i$
- $\Rightarrow$  затруднения в корректировке при назначении нового старосты + расход памяти для хранения дублированной  $\dot{i}$

# ТИПЫ СВЯЗЕЙ

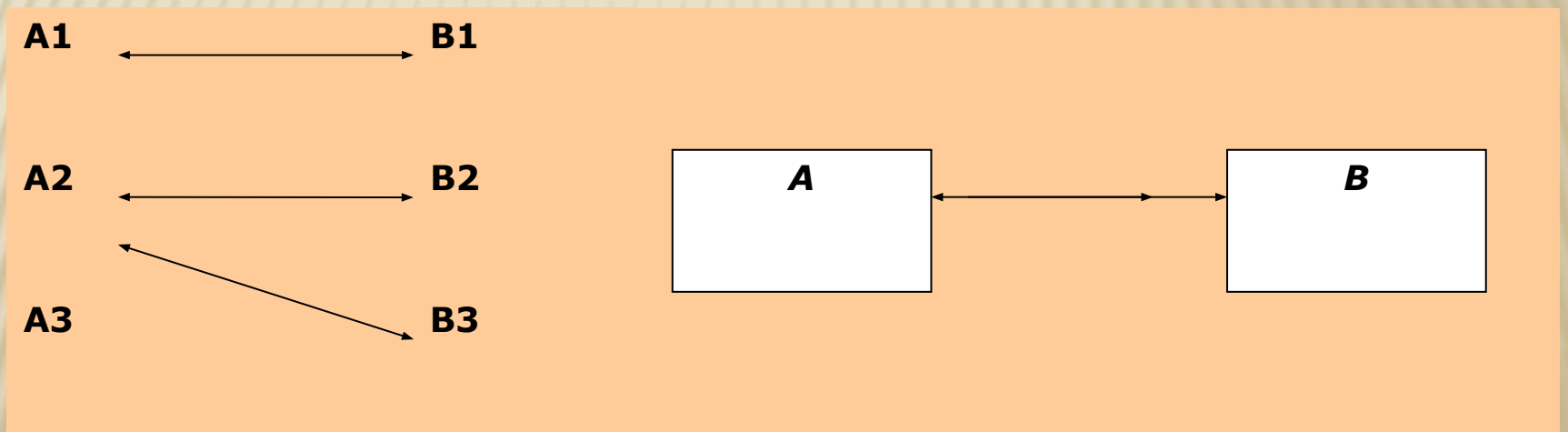
## «ОДИН К ОДНОМУ»

В каждый момент времени 1-экземпляру объекта *A* соответствует не более 1-объекта *B* и наоборот



# «ОДИН КО МНОГИМ»

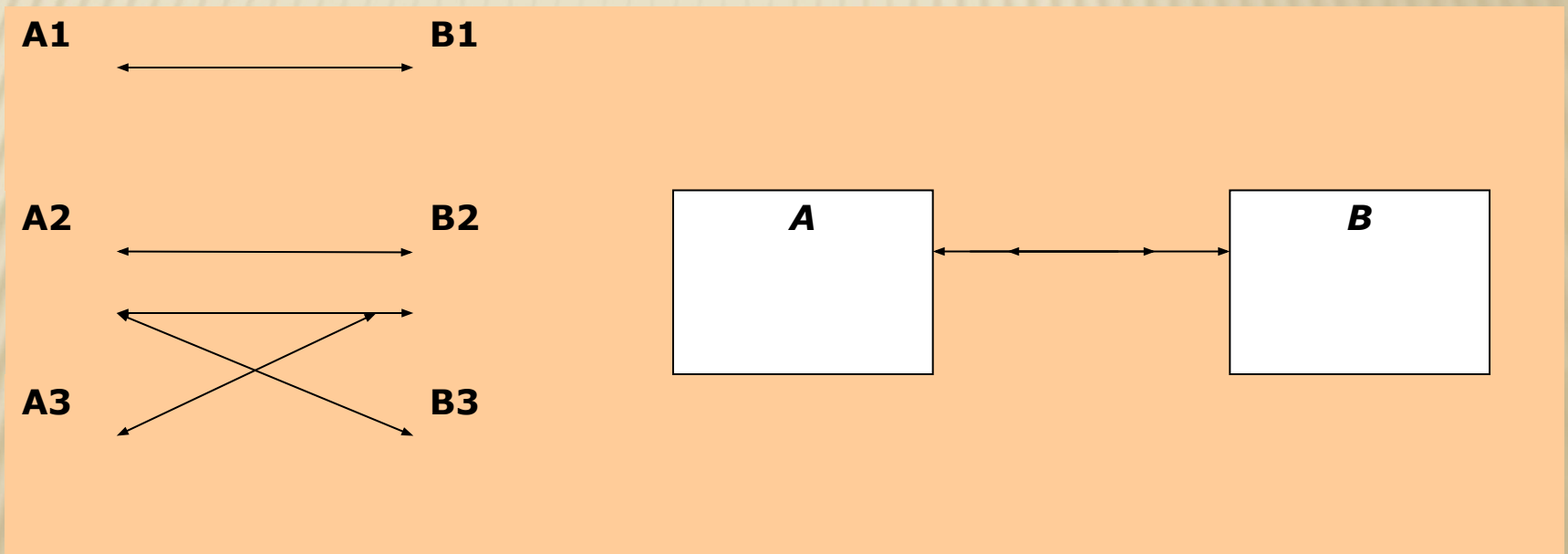
1-объекту  $A$  соответствует 0, 1 экземпляров  $B$ , но не более





# «МНОГИЕ КО МНОГИМ»

1-экземпляру объекта *A* соответствует 0,1 или более экземпляров *B* и наоборот



# АТТРИБУТЫ

---

≡ атрибут(ы) со значением, уникальным для любого  $\hat{o}$  данного типа

⇒ любой  $\hat{o}$  будет однозначно определен значением первичного ключа

Потенциальные ключи ≡ комбинация(и) атрибутов

∇ м.б. использован в качестве первичного ключа (на схеме подчеркивается)

□ Первичный ключ  $\hat{o}$  ЧЕЛОВЕК = № карточки соц. страхования НКСС □

# ВИДЫ АТТРИБУТОВ

---

Составные атрибуты  $\equiv$  образованы неск. атрибутами  
(на схеме соединяются прямой)

□  $\hat{o}$  Человек: два составных атрибута (Имя, Адрес) □

Многозначный атрибут  $\equiv$  имеет не единственное  
значение (выделяется двойным контуром)

□ Адрес является многозначным:  $\hat{o}$  Человек могут  
иметь несколько адресов □

# ПРОИЗВОДНЫЕ АТТРИБУТЫ

---

- ≡ принимают значения, которые могут быть вычислены по другим атрибутам того же типа  $\hat{o}$  или по атрибутам типов, с которыми данный тип  $\hat{o}$  может быть связан отношением (обозначаются заливкой)
- значение Транспортный налог для  $\hat{o}$  МАШИНА м.б. получено по атрибуту Количество цилиндров □

# ТИПЫ ОТНОШЕНИЙ У АТТРИБУТОВ

---

- ∃ при наличии связей между типами  $\hat{O}$
- Дата покупки не м.б. атрибутом для типа Человек, т.к. в этом случае  $\hat{O}$  Человек смогут владеть лишь одной машиной □
- Не имеет смысла приписывать этот атрибут и типу Машина, который был куплен-продан несколько раз, и непонятно, кто купил машину и когда □