



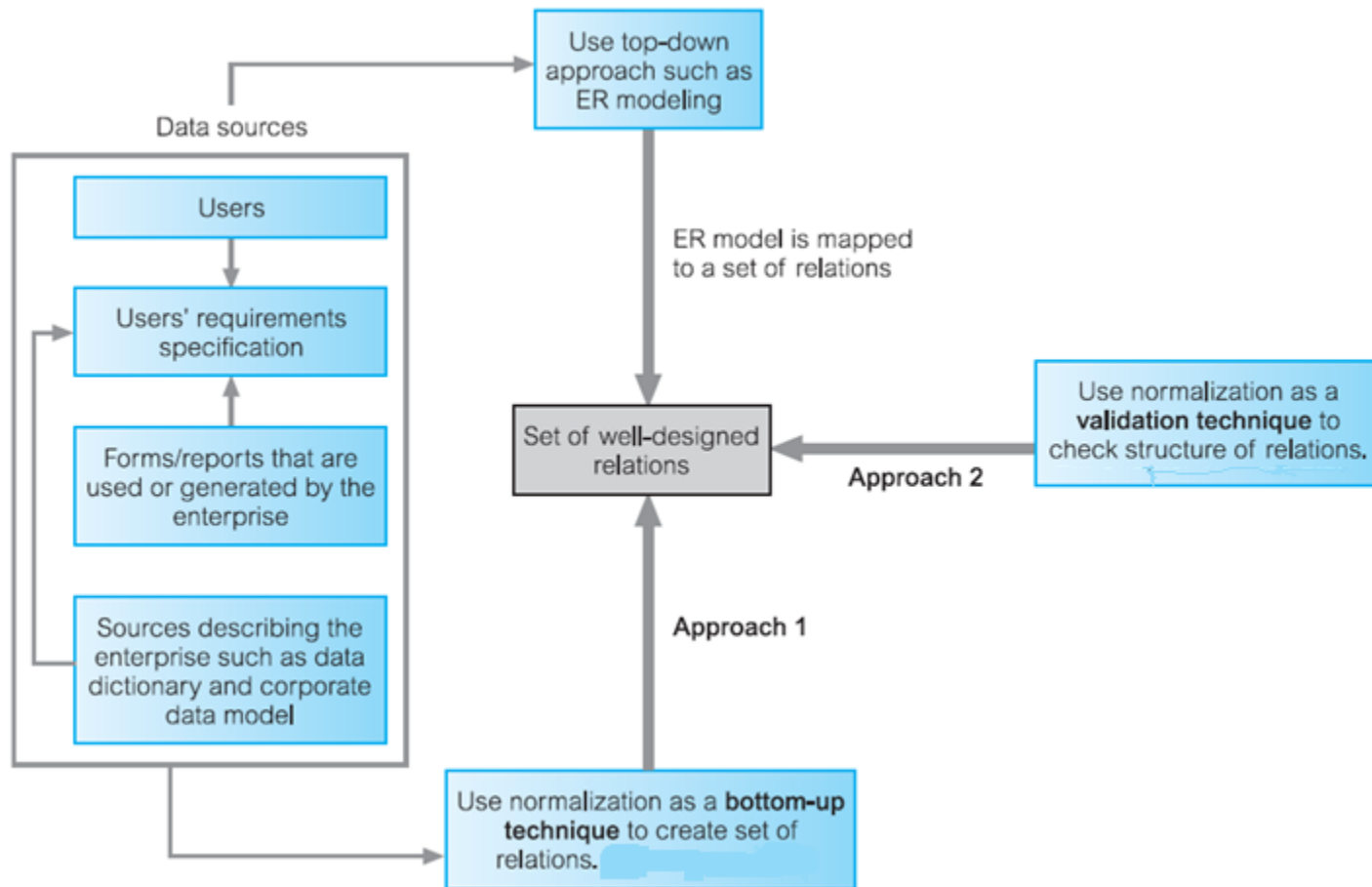
Бази от данни

Релационен модел на данни. Функционални
зависимости

Дизайн на БД. Подходи

- ▶ Както видяхме в предишните лекции, релационната схема на базата от данни може да бъде получена, чрез преобразуване на E/R модела към релационен.
- ▶ Друг начин да бъде получена релационна схема е дизайнера на базата от данни, да я създаде направо съобразявайки се със съответните бизнес изисквания и информацията, която има за данните.
- ▶ При лош дизайн, вторият начин може да доведе до лошо моделирана база от данни, която да позволява излишества, повторения и неконсистентност на данните.
- ▶ И в двата случая получените релационни схеми могат да бъдат подобрени посредством ограничения наречени – функционални зависимости (ФЗ).
- ▶ Функционалните зависимости – описват връзката между атрибутите на релацията.

Дизайн на БД. Подходи



Книга: Database systems. A practical approach to design, implementation and management,
Автори: Thomas Connolly, Carolyn Begg

Дизайн на БД. Подходи

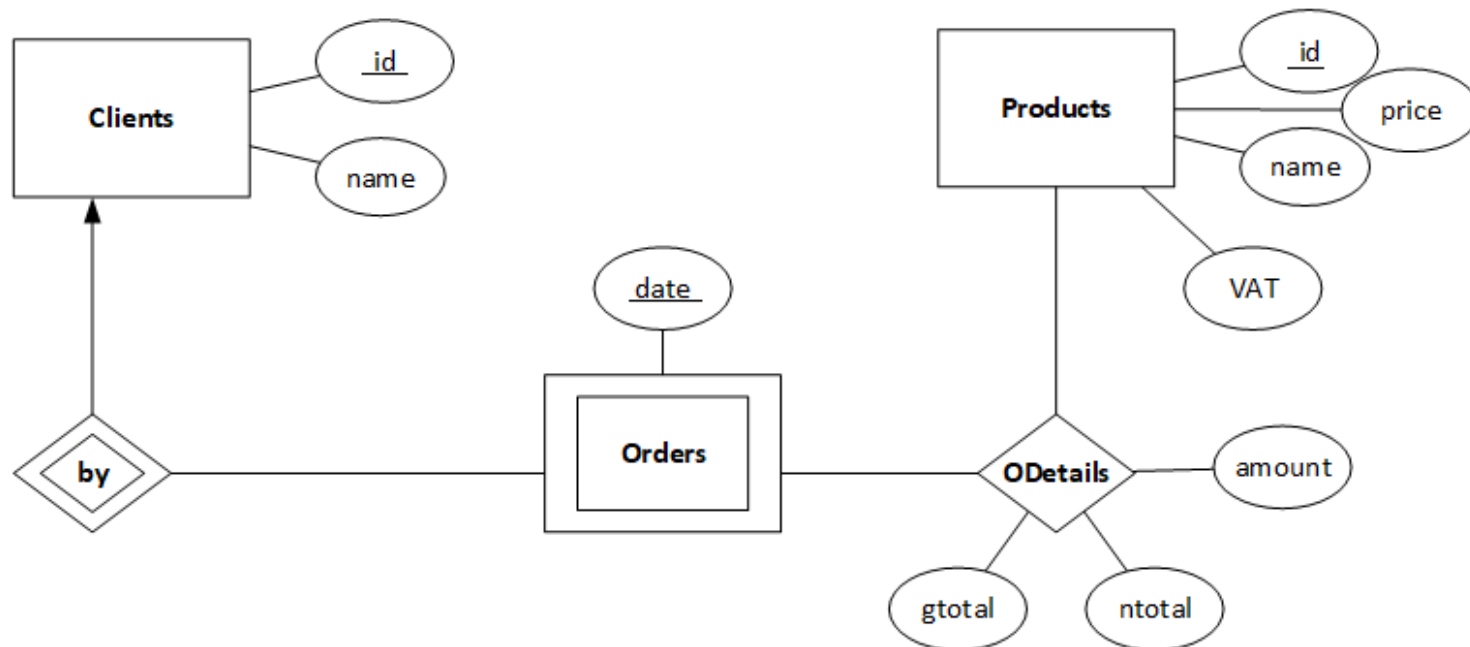
- ▶ **Подход 1:** Правим E/R модел по зададената ни информация и преобразуваме до релационни схеми. Намираме всички ФЗ за релациите и нормализираме, ако е необходимо.
- ▶ **Подход 2:** Поставяме всички атрибути (характеристики) на данните в една релация. Намираме всички ФЗ за релацията. Нормализираме.
- ▶ **Нормализация** наричаме процеса на декомпозиране на релация на няколко релации с цел отстраняване на аномалии и повторения.

Задача

- ▶ Да се направи дизайн на БД, съдържаща информация за поръчки.
Известно ни е, че в БД ще се пази информация за дата на поръчката, номер на продукт – уникален за всеки продукт, име на продукт, цена на продукт, ДДС за съответния продукт, номер на клиент – уникален за всеки клиент, име на клиент, количество, което клиентът е поръчал от продукта в съответната поръчка, сума на поръчката без ДДС, сума на поръчката с ДДС.
- ▶ В сила са следните предположения:
 - ▶ Стойността на ДДС за продукта може да варира за различните продукти.
 - ▶ Поръчките на клиентите, направени в един и същ ден (на една и съща дата) се обединяват, т.е. има само една поръчка за клиент за дадена дата.
 - ▶ $\text{сума на поръчката с ДДС} = \text{сума на поръчката без ДДС} + \text{ДДС}$
- ▶ 1. Направете модел на БД
- ▶ 2. Определете ключовете за релациите (релацията).

Решение – Подход 1

Подход 1: Правим E/R модел по описаната информация и преобразуваме до релационни схеми. Намираме всички ФЗ за релациите и нормализираме, ако е необходимо.



Преобразуваме:

- ▶ Clients(cID, cName);
- ▶ Products(pID, pPrice, pName, VAT);
- ▶ **Orders(cID, oDate)**
- ▶ ODetails(cID, oDate, pID, amount, ntotal, gTotal)

Подход 2

- ▶ Поставяме всички атрибути (характеристики) на данните в една релация. Намираме всички ФЗ за релацията. Нормализираме.
- ▶ За да разберем Подход 2, трябва да дефинираме:
 - ▶ Функционални зависимости (ФЗ)
 - ▶ Правила за ФЗ
 - ▶ Покритие на множество от ФЗ
 - ▶ Връзка между покритие на множество от ФЗ и първичен ключ (РК)
 - ▶ Нормални форми и декомпозиране на релации

Релационен модел на данни

- ▶ **Релационният модел** представя данните в двумерни таблици наречени релации.
- ▶ **Схема на базата от данни** – наричаме формалното описание на релациите в базата и връзките между тях.
- ▶ **Атрибути** на релация са имената на колоните. Те описват съдържанието на колоните.
- ▶ **Схема на релация** – наричаме името на релацията последвана от списък с атрибутите към релацията. Например $R(A_1, A_2, A_3, \dots, A_n)$.
- ▶ **Кортежи** – Редовете на релацията се наричат кортежи. Кортежите имат по една стойност за всеки атрибут от релацията.
- ▶ **Домейн** наричаме множеството от допустими стойности за даден атрибут. Домейна е част от релационната схема.

Релационен модел на данни - Пример

- ▶ **Схема на релация:**

MovieStar(name, address, gender, birthdate)

- ▶ **Атрибути на релация:**

name, address, gender, birthdate

- ▶ **Домейн:**

- ▶ birthdate: date
- ▶ gender: string

- ▶ MovieStar(name: string, address: string, gender: string, birthdate: date)

- ▶ **Кортеж:**

('Jane Fonda', 'Turner Av. ', 'F', '1977-07-07')

	MOVIESTAR
	NAME
	ADDRESS
	GENDER
	BIRTHDATE

Функционални зависимости - определение

- ▶ Нека $R(A, B)$ е релация. Функционална зависимост (ФЗ) на релацията R , наричаме твърдение от вида $A \rightarrow B$ (чете се атрибута A на релацията R функционално (еднозначно) определя атрибута B), т.е. за всеки два кортежа от релацията R е изпълнено, че ако те съвпадат по атрибута A , то те съвпадат и по атрибута B
- ▶ В обща случай ФЗ, могат да бъдат записани така: $A_1, A_2, \dots, A_n \rightarrow B$, където A_i, B са атрибути на релацията R
- ▶ Ключът на една релация, функционално определя всеки атрибут в релацията. Или казано по друг начин, ако в една релация два кортежа имат едни и същи стойности по ключовите атрибути, то те са един и същ кортеж.

Правило за разделяне и комбиниране

- ▶ Правило за разделяне: Ако е в сила ФЗ $A_1, A_2, \dots, A_n \rightarrow B_1, B_2, \dots, B_m$ за релацията R , то в сила са и функционалните зависимости:

$$A_1, A_2, \dots, A_n \rightarrow B_1$$

$$A_1, A_2, \dots, A_n \rightarrow B_2$$

$$A_1, A_2, \dots, A_n \rightarrow B_m$$

- ▶ Правило за комбиниране: Ако са в сила ФЗ:

$$A_1, A_2, \dots, A_n \rightarrow B_1$$

$$A_1, A_2, \dots, A_n \rightarrow B_2$$

$$A_1, A_2, \dots, A_n \rightarrow B_m$$

или множеството от атрибути A_1, A_2, \dots, A_n функционално определя повече от един атрибут, то в сила е и ФЗ $A_1, A_2, \dots, A_n \rightarrow B_1, B_2, \dots, B_m$

Правило за разделяне и комбиниране

- ▶ Можем да разделяме атрибутите в дясната страна, така че само един атрибут да се появява от дясно за всяка функционална зависимост (правило за разделяне).
- ▶ Също така можем да заменим колекцията от ФЗ в текущата лява страна с една функционална зависимост получена от комбинирането на десните страни (правило за комбиниране)
- ▶ Множествата от ФЗ, които се получават са еквивалентни.
- ▶ Правилото за разделяне и комбиниране са приложими само за десните страни на ФЗ. Ако приложим тези правила за левите страни ново-получените ФЗ няма да са в сила за съответната релация.
- ▶ **Тривиални функционални зависимости** – една функционална зависимост наричаме тривиална, ако $A_1, A_2, \dots, A_n \rightarrow B$ и атрибута B е един от атрибутите A_i в лявата част
- ▶ Тривиалните ФЗ е все едно да се каже, че ако два кортежа съвпадат по A_1 и A_2 , то те съвпадат по A_1 .

Правило за тривиална ФЗ

- ▶ В оригиналната дефиниция на ФЗ, се изключват тривиалните ФЗ (не се позволяват тривиални ФЗ)
- ▶ Но позволяването на тривиални ФЗ, които по дефиниция винаги са в сила към ФЗ не би навредило. Понякога тривиалните ФЗ опростяват твърденията на правилата.
- ▶ **Дефиниция:** Казваме , че ФЗ $A_1, A_2, \dots, A_n \rightarrow B_1, B_2, \dots, B_m$ е
 - ▶ Тривиална – ако множеството от $\{ B_j \mid j = 1, 2, 3, \dots, m \} \subseteq \{ A_i \mid i = 1, 2, 3, \dots, n \}$
 - ▶ Нетривиална – ако съществува поне едно B_j , което да е различно от A_i
 - ▶ Напълно нетривиална – ако всички B_j са различни от A_i
- ▶ Винаги можем да премахнем от дясната страна на ФЗ, тези атрибути които се появяват в ляво. Така ФЗ: $A_1, A_2, \dots, A_n \rightarrow B_1, B_2, \dots, B_m$ е еквивалентна на $A_1, A_2, \dots, A_n \rightarrow C_1, C_2, \dots, C_r$, където C -тата са всички тези B -та, които не са A -та. Т.е. махнати са тривиалните ФЗ. Това правило наричаме правило за тривиалните ФЗ

Пример за ФЗ

- ▶ Нека разгледаме релацията:

`StarMovies(title, year, length, filtype, studioName, starName)`

- ▶ Обърнете внимание, че в релацията `StarMovies` за разлика от релацията `Movies` има добавен допълнителен атрибут `starName` за името на актьора. Това може да породи повторение на информация и излишества за релацията `StarMovies`.
- ▶ **Задача:** Да се определят ФЗ за релацията `StarMovies`
- ▶ **Решение:**
- ▶ Известно ни е че ключът за релацията `Movies` е `title, year`.
- ▶ Като приложим правилото за комбиниране от предишния слайд, получаваме:

`title, year -> length, filmType, studioName (ФЗ-1)`

Пример за функционални зависимости

- ▶ Тази ФЗ: $\text{title, year} \rightarrow \text{length, filmType, studioName}$, ни казва че ако два кортежа от релацията **StarMovies** имат еднакви стойности за **title** и **year**, то те ще имат и еднакви стойности и за **length**, **filmType** и **studioName**.
- ▶ Т.е по зададени **title, year** се определя еднозначно филм. По тези причини и на конкретен филм по зададени **title, year** еднозначно ще съответстват **length, filmType, studioName**
- ▶ За атрибута **starName**, обаче това не е така.
- ▶ **ФЗ $\text{title, year} \rightarrow \text{starName}$, не е в сила за релацията **StarMovies****
- ▶ Причината е, че ако два кортежа съвпадат по **title, year** това не означава, че трябва да съвпадат и по **starName**. В един филм може да участват много актьори и така на един **title, year** може да съответстват няколко имена на актьори.

Ключове на релациите

- ▶ Множеството от един или няколко атрибута $\{A_1, A_2, \dots, A_n\}$ на релацията R , наричаме ключ за релацията ако:
 1. $\{A_1, A_2, \dots, A_n\}$ функционално определят всичко останали атрибути на R
 2. Множеството от атрибути $\{A_1, A_2, \dots, A_n\}$ е минимално, т.е. не съществува под-множество на $\{A_1, A_2, \dots, A_n\}$, за което да е изпълнено първото условие
- ▶ Първото изискване за ключ е множеството от атрибути функционално да определя всички атрибути на R . Тъй като релациите са множества – не е възможно за два различни кортежи от R да съвпадат по всичките си атрибути. Ако това е така, то значи става въпрос за един и същ кортеж.
- ▶ Пример: Нека е дадена релацията $R(A, B, C)$ и ни е известно, че са в сила следните ФЗ: $A \rightarrow B$ и $A \rightarrow C$. Атрибутът A функционално определя всички атрибути на R , следователно е ключ за релацията R . Обратно, ако A е ключ за R , то може да твърдим, че са в сила следните ФЗ: $A \rightarrow B$ и $A \rightarrow C$

Ключове на релациите - пример

- ▶ Нека разгледаме отново релацията:

`StarMovies(title, year, length, filtype, studioName, starName)`

- ▶ Известно ни е че за релацията `StarMovies` е в сила следната ФЗ:

`title, year -> length, filmType, studioName (Ф31)`

- ▶ Твърдение: Ключът за релацията е `{title, year, starName}`. Защо?
 - ▶ Тези три атрибута – взети заедно, функционално определят всички останали атрибути.
 - ▶ Нека допуснем, че два кортежа от `StarMovies` съвпадат по `title, year, starName`. Понеже съвпадат по `title, year`, то от Ф31 те ще съвпадат и по `length, filmType` и `studioName`. Т.е тези два кортежа съвпадат по всичките атрибути на релацията, следователно са един и същ кортеж (релациите са множества и не допускат повторение на кортежи). Следователно трите атрибута `{title, year, starName}` функционално определят всеки атрибут в `StarMovies`.
- ▶ Възможните подмножества на `{title, year, starName}` са:
 - ▶ `{title, year}` – не определя функционално `starName`
 - ▶ `{title, starName}` – не определя функционално `year`
 - ▶ `{year, starName}` – не определя функционално `title`

Функционални зависимости и схема

- ▶ Функционалните зависимости, както всяко ограничение са една проверка за релационната схема, а не за инстанция на релацията
- ▶ Това е ограничение, на което всички кортежи на релацията трябва да отговарят
- ▶ Както и при ограниченията, това че кортежите съвпадат в някои колони не прави тези колони функционално зависими, защото е възможно добавянето на нов кортеж, които да наруши предполагаемата функционална зависимост
- ▶ Понякога релациите имат повече от един възможен ключ. В тези случаи един от ключовете се обозначава като първичен ключ. Изборът на първичен ключ е важен, защото за някои БД, той може да определи как се записват данните на диска. Важна конвенция, която трябва да се спазва е да се подчертават атрибутите на първичния ключ в релационната схема.
- ▶ Един ключ е минимален, ако от него не можем да изключим атрибут. Това съвсем не означава, че атрибутите принадлежащи на ключа трябва да са минимален брой. Може да имаме повече от един минимален ключ.
- ▶ Функционалните зависимости се наричат функционални, защото на даден списък от аргументи A_1, A_2, \dots, A_n се съпоставя единствена стойност B , т.е. както при функциите.

Суперключ

- ▶ Множеството от атрибути, което съдържа ключа се нарича суперключ.
- ▶ Всеки суперключ удовлетворява първото условие за ключ – той функционално определя всички атрибути на релацията. За суперключа не се изисква да удовлетворява второто условие - за минималност на ключа.

- ▶ Нека разгледаме релацията

`StarMovies(title, year, length, filtype, studioName, starName)`

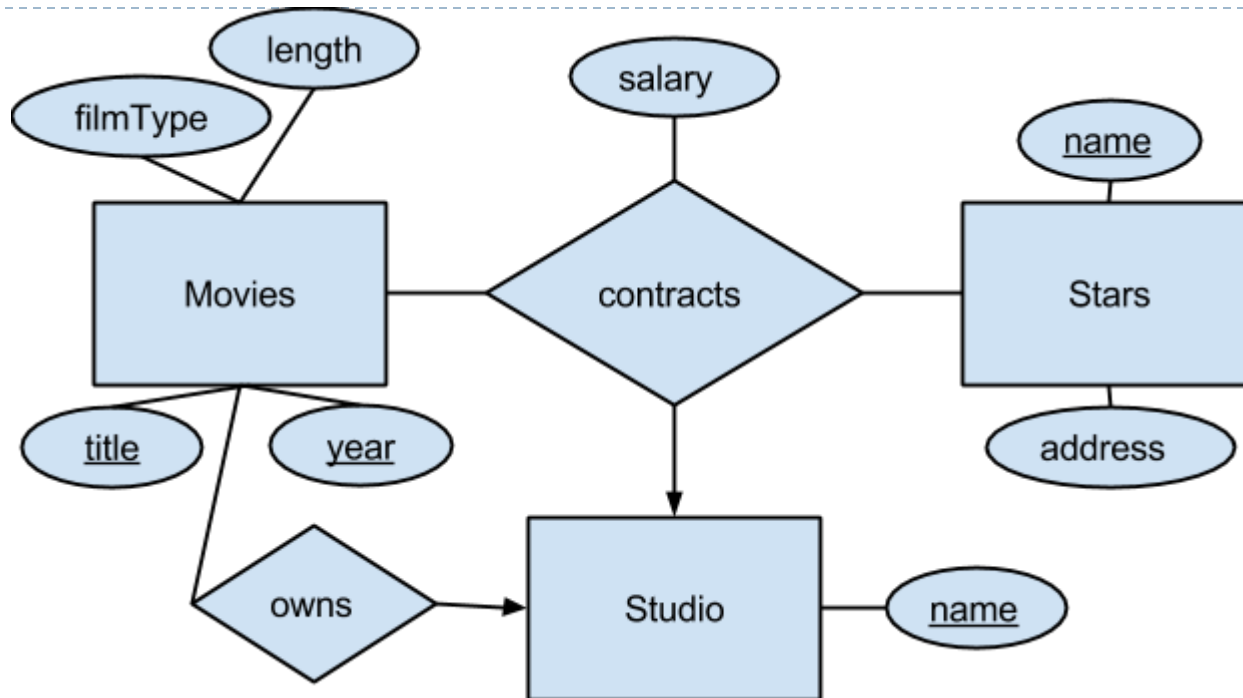
- ▶ Суперключ за горната релация е:

`{title, year, starName, length}` или `{title, year, starName, studioName}`

Откриване на ключовете на релациите – Подход 1

- ▶ Когато преобразуваме E/R диаграма към релационна схема на БД има правила, как да определим ключовете за релациите:
 1. Ако релацията е получена в следствие на преобразуване на множество от същности, то тогава ключът на релацията е ключът на това множество от същности.
 2. Ако релацията R е получена от връзка, тогава в зависимост от вида на връзката се определя и кой ще бъде ключа на R
 - ▶ Ако връзката е много-много, тогава ключовете на двете свързани множества същности са ключовите атрибути за релацията R
 - ▶ Ако връзката е много-един от множеството от същности E1 към множеството от същности E2, тогава ключовите атрибути на E1 са ключовите атрибути на R
 - ▶ Ако връзката е едно-едно, тогава ключовите атрибути на едно от свързаните множества са ключови атрибути за R. Тук не може еднозначно да се каже кой от двата ключа на E1 или на E2, ще бъде ключа.
 - ▶ Ако връзката R е небинарна и свързва множествата от същности E1, E2 и E3 и има стрелка към множеството от същности E1, то тогава ключът за съответстващата релация се образува от ключовете на другите множества без ключовете на E1.

Откриване на ключовете на релациите - Пример



- ▶ Movies(title, year, length, filmType)
- ▶ Studio(name)
- ▶ Stars(name, address)
- ▶ Contracts(title, year, starName, studioName, salary)
- ▶ Owns(title, year, studioName)

Намиране на ключ – Подход 2

- ▶ Нека релацията R има атрибути – A, B, C, D, E и F
- ▶ Имаме и множеството от ФЗ, които са в сила за R
 1. $AB \rightarrow C$
 2. $BC \rightarrow AD$
 3. $D \rightarrow E$
 4. $CF \rightarrow B$
- ▶ Искаме да намерим ключа за R ?

Покритие на атрибути

- ▶ **Дефиниция:** Нека $R(A_1, A_2, B_1, B_2, B_3)$, а S е множество от ФЗ в сила за R . Покритие на множеството $\{A_1, A_2\}$, наричаме множеството от атрибутите $\{B_i \mid i = 1, 2, 3\}$ на R , за които е в сила че ФЗ: $A_1, A_2 \rightarrow B_i$ може да бъде получена от ФЗ на S .
- ▶ Покритие на множество атрибути означаваме с $\{A_1, A_2, A_3, \dots, A_n\} +$
- ▶ За да улесним пресмятането на покритие, ще позволим тривиалните ФЗ.

Алгоритъм за пресмятане на покритие

1. Дадена е $R(A_1, A_2, B_1, B_2, B_3)$ и множество от ФЗ - S . Търсим покритието на $\{A_1, A_2\}$
2. Нека $X = \{A_1, A_2\}$
3. Търсим ФЗ от вида $A_1, A_2 \rightarrow B_i$, от множеството S , такива че, левите страни на ФЗ са с атрибути, които принадлежат на множеството X , а атрибута B_i не.
4. Добавяме атрибута B_i към множеството X .
5. Повтаряме стъпка 3 и 4 докато стигнем ситуация, в която не можем да добавим повече нови атрибути към X . Понеже множеството X , може само да нараства, а броят на атрибутите е краен ще достигнем до момент, в който няма да можем да добавим нищо повече към X .
6. Множеството X е търсеното покритие.

Покритие и ключове

- ▶ Нека $R(A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m)$
- ▶ Покритието $\{A_1, A_2, \dots, A_n\}^+$ на релацията R е множество от всичките атрибути на релацията, тогава и само тогава когато $\{A_1, A_2, \dots, A_n\}$ е супер-ключ на R , т.е. функционално определя всеки атрибут на R .
- ▶ Ако $\{A_1, A_2, \dots, A_n\}$ е супер-ключ то ще бъдат в сила ФЗ от вида:
 - ▶ $A_1, A_2, \dots, A_n \rightarrow B_1, B_2, \dots, B_m$
 - ▶ Следователно B_i ще принадлежат на покритието на $\{A_1, A_2, \dots, A_n\}^+$
 - ▶ Следователно покритието $\{A_1, A_2, \dots, A_n\}^+$ на релацията R е множество от всичките атрибути на релацията
- ▶ Обратно, ако покритието $\{A_1, A_2, \dots, A_n\}^+$ съдържа всички атрибути на релацията R , от тук ще следва че всеки един атрибут принадлежащ на множеството на покритието е функционално определен от $\{A_1, A_2, \dots, A_n\}$. Следователно $\{A_1, A_2, \dots, A_n\}$ е супер-ключ за R .

Пример за пресмятане на покритие

- ▶ Нека релацията R има атрибути – A, B, C, D, E и F
- ▶ Имаме и множеството от ФЗ, които са в сила за R
 1. $AB \rightarrow C$
 2. $BC \rightarrow AD$
 3. $D \rightarrow E$
 4. $CF \rightarrow B$
- ▶ Искаме да намерим ключа за R ?
- ▶ Намираме покритието на множеството $\{A, B\}$?
- ▶ По алгоритъма:
 1. $X = \{A, B\}$
 2. Разширяваме X , като добавяме нови атрибути според ФЗ. От $AB \rightarrow C$ добавяме C към X . От $BC \rightarrow AD$, добавяме D към X . От $D \rightarrow E$ добавяме E към X . От последната ФЗ, не можем да добавим нищо към X , така покритието на $\{A, B\}$ е множеството $\{A, B, C, D, E\}$
- 1. Ключът на R е $\{A, B, F\}$

Пример за пресмятане на покритие

- ▶ Нека релацията R има атрибути – A, B, C, D, E и F
- ▶ Имаме и множеството от ФЗ, които са в сила за R
 1. $AB \rightarrow C$
 2. $BC \rightarrow AD$
 3. $D \rightarrow E$
 4. $CF \rightarrow B$
- ▶ Искаме да проверим дали R има други кандидат-ключове?
- ▶ Намираме покритието на множеството $\{B, C\}, \{D\}, \{CF\}$?
 1. Кандидат-ключове за R са $\{B, C, F\}, \{C, F\}$

Задача

- ▶ Да се направи дизайн на БД, съдържаща информация за поръчки.
Известно ни е, че в БД ще се пази информация за дата на поръчката, **номер на продукт – уникален за всеки продукт**, име на продукт, цена на продукт, ДДС за съответния продукт, **номер на клиент – уникален за всеки клиент**, име на клиент, количество, което клиентът е поръчал от продукта в съответната поръчка, сума на поръчката без ДДС, сума на поръчката с ДДС.
- ▶ В сила са следните предположения:
 - ▶ Стойността на ДДС за продукта може да варира за различните продукти.
 - ▶ Поръчките на клиентите, направени в един и същ ден (на една и съща дата) се обединяват, т.е. има само една поръчка за клиент за дадена дата.
 - ▶ $\text{Сума на поръчката с ДДС} = \text{сума на поръчката без ДДС} + \text{ДДС}$
- ▶ 1. Направете модел на БД
- ▶ 2. Определете ключовете за релациите (релацията).

Решение – Подход 2

Подход 2: Поставяме всички атрибути (характеристики) на данните в една таблица. Намираме всички ФЗ за тази релация. Намираме кандидат-ключовете. Нормализираме.

- ▶ Поставяме всичко в една релация

Order(pld, pName, pPrice, Amount, cld, cName, oDate, VAT, nTotal, gTotal)

- ▶ Намираме всички ФЗ за релацията Order (от дадената ни информация)
 - ▶ ФЗ_1: pld -> pName, pPrice, VAT
 - ▶ ФЗ_2: cld -> cName
 - ▶ ФЗ_3: pld, cld, oDate -> Amount
 - ▶ ФЗ_4: pPrice, Amount -> nTotal
 - ▶ ФЗ_5: nTotal, VAT -> gTotal
 - ▶ ФЗ_6: cld, oDate -> pld (това НЕ е вярно - защо?)

Намиране на ключ – Подход 2

- ▶ Дадена е релацията:

Order (pId, pName, pPrice, Amount, cId, cName, oDate, VAT, nTotal, gTotal)

- ▶ **В сила са следните ФЗ:**

- ▶ ФЗ_1: pId -> pName, pPrice, VAT
- ▶ ФЗ_2: cId -> cName
- ▶ ФЗ_3: pId, cId, oDate -> Amount
- ▶ ФЗ_4: pPrice, Amount -> nTotal
- ▶ ФЗ_5: nTotal, VAT -> gTotal

- ▶ **Задача:** Намерете ключа за Order

- ▶ **Решение:** Намираме покритието на {pId, cId, oDate}

{ pId, pName, pPrice, Amount, cId, cName, oDate, VAT, nTotal, gTotal }

- ▶ Следователно {pId, cId, oDate} е суперключа за Order
- ▶ {pId, cId, oDate} ключ ли е за Order? – ако премахнем pId или cId или oDate?

Задача

- Дадена е релацията Invoice

INV_NUM	PROD_NUM	SALE_DATE	PROD_DESCRIPTION	VEND_CODE	VEND_NAME	NUMBER_SOLD	PROD_PRICE
12345	EE-A3422Z	22_JUL_2020	Camera	502	City Shop	5	\$69.95
12345	AC-2002X	22_JUL_2020	Cable	502	City Shop	6	\$3.45
12345	SV-4534Y	22_JUL_2020	Keyboard	609	COVID-19 Shop	5	\$39.99
12346	EE-A3422Z	22_JUL_2020	Camera	502	City Shop	3	\$69.95
12347	BC-7783P	23_JUL_2020	Monitor	257	Online Shop	5	\$87.75

Invoice (inv_num, prod_num, sale_date, prod_desc, vend_code, vend_name, number_sold, prod_price)

Задача

- ▶ Определете ФЗ за релацията. Намерете ключа за релацията.
- ▶ Решение:
- ▶ В сила са следните ФЗ за релацията Invoice:
 - ▶ `prod_num` -> `prod_desc`, `prod_price`
 - ▶ `vend_code` -> `vend_name`
 - ▶ `inv_num` -> `sale_date`
 - ▶ `prod_num`, `inv_num` -> `number_sold`
- ▶ Ключът за релацията е `{prod_num, vend_code, inv_num}`

Задача

- Дадена е релацията StarWars

<u>CharId</u>	<u>CharName</u>	<u>ActorName</u>	<u>FigId</u>	<u>OwnerId</u>	Pseudonym
Vader	Darth Vader	David Prowse	f14	Bill	Jabba the Hoot
Vader	Darth Vader	David Prowse	f22	Amy	Don't Blame Me
Yoda	Yoda	Frank Oz	f16	Lucy	<u>Xena</u> Warrior
Leia	Princess Leia	Carrie Fisher	f45	Bill	Jabba the Hoot
Leia	Princess Leia	Carrie Fisher	f99	Amy	Don't Blame Me

StarWars (charid, charname, actorname, figid, ownerid, pseudonym)

Задача

- ▶ Определете ФЗ за релацията. Намерете ключа за релацията.
- ▶ Решение:
- ▶ В сила са следните ФЗ за релацията StarWars :
 - ▶ figid -> ownerid, pseudonym
 - ▶ charid -> charname, actorname
- ▶ Ключът за релацията е {figid, charid}

Покритие и ФЗ

- ▶ Ако знаем как да намерим покритието на дадено множество от атрибути ще можем да проверим и дали дадена ФЗ може да бъде изведена от множеството от ФЗ, което имаме за R (множеството S).
- ▶ Например нека ни е дадена ФЗ: $A_1, A_2, \dots, A_n \rightarrow B$. Искаме да проверим дали тази ФЗ може да бъде изведена (функционално следва) от ФЗ на S . Това става по следния начин:
 1. Построяваме покриващото множество за атрибутите $\{A_1, A_2, \dots, A_n\}$, като използваме ФЗ на S
 2. Ако атрибутът B принадлежи на покритието, тогава ФЗ: $A_1, A_2, \dots, A_n \rightarrow B$ логически следва от S
 3. Ако атрибутът B не принадлежи на покритието, тогава ФЗ: $A_1, A_2, \dots, A_n \rightarrow B$ не следва от S
- ▶ По общо казано, ФЗ: $A_1, A_2, \dots, A_n \rightarrow B_1, B_2, \dots, B_m$ следва от множеството ФЗ S тогава и само тогава, когато $\{B_i\}$ принадлежат на покритието на $\{A_1, A_2, \dots, A_n\}^+$

Покритие и ФЗ - Пример

- ▶ Нека релацията R има атрибути – A, B, C, D, E и F
- ▶ Имаме и множеството от ФЗ S , които са в сила за R
 1. $AB \rightarrow C$
 2. $BC \rightarrow AD$
 3. $D \rightarrow E$
 4. $CF \rightarrow B$
- ▶ Искаме да проверим дали ФЗ: $AB \rightarrow D$ логически следва от S ?
 1. По алгоритъма намираме, че покритието на $\{A, B\}$ е $X = \{A, B, C, D, E\}$
 2. Понеже атрибута D принадлежи на множеството X , то ФЗ: $AB \rightarrow D$, може да бъде изведена от S (т.е. Също е в сила за R).
- ▶ Да разгледаме друга ФЗ: $D \rightarrow A$
 1. По алгоритъма намираме, че покритието на $\{D\}$ е $X = \{D, E\}$
 2. Понеже атрибута A не принадлежи на множеството X , то ФЗ: $D \rightarrow A$, **не** може да бъде изведена от S , (т.е. Не е в сила за R).

Аксиоми на Армстронг

Аксиомите на Армстронг са правила, по които можем да генерираме нови ФЗ, които също са в сила за съответната релация. По-долу са основните аксиоми на Армстронг. Нека е дадена релацията $R(A_1, A_2, B_1, B_2)$

Основни:

1. **Рефлексивност (reflexivity):** Ако $\{B_1, B_2\}$ е под-множество на $\{A_1, A_2\}$, то $A_1, A_2 \rightarrow B_1, B_2$
2. **Умножение (augmentation):** Ако $A_1, A_2 \rightarrow B_1$ то и $A_1, A_2, B_2 \rightarrow B_1, B_2$
3. **Транзитивност (transitivity):** Ако $A_1, A_2 \rightarrow B_1$ и $B_1 \rightarrow B_2$, то $A_1, A_2 \rightarrow B_2$
4. **Добавяне (composition):** Ако $A_1 \rightarrow B_1$ и $A_2 \rightarrow B_2$, то $A_1, A_2 \rightarrow B_1, B_2$

Рефлексивност - Доказателство

Рефлексивност: Ако $\{B1, B2\}$ е под-множество на $\{A1, A2\}$, то $A1, A2 \rightarrow B1, B2$

Доказателство: Това са тривиални ФЗ.

Умножение - Доказателство

Умножение: Ако $A1, A2 \rightarrow B1$ то и $A1, A2, B2 \rightarrow B1, B2$

Доказателство:

Да намерим покритието на $\{A1, A2, B2\}$.

- ▶ По алгоритъма, на първа стъпка в множеството на покритието попадат атрибутите $A1, A2, B2$
- ▶ Но $A1, A2 \rightarrow B1$, следователно от 2 стъпка на алгоритъма в множество ще попадне и $B1$
- ▶ Така покритието на $\{A1, A2, B2\}$ се състои от атрибутите $A1, A2, B1, B2$, следователно $A1, A2, B2$ функционално определя всеки атрибут принадлежащ на покритието
- ▶ Следователно $A1, A2, B2 \rightarrow B1, B2$

Транзитивност - Доказателство

- ▶ **Твърдение:** Ако $A1, A2 \rightarrow B1$ и $B1 \rightarrow B2$, то $A1, A2 \rightarrow B2$

Доказателство:

- ▶ Намираме покритието на $\{A1, A2\}$
- ▶ От първата ФЗ следва, че $B1$ принадлежат на покритието.
- ▶ От втората ФЗ, следва че и $B2$ принадлежат на покритието.
- ▶ От тук следва, че $A1, A2 \rightarrow B2$ е вярна за релацията ФЗ

Проектиране на ФЗ

- ▶ Да допуснем, че имаме релацията $R1$, за която са в сила множество от ФЗ – S . Искаме да направим проекция (да премахнем атрибути) на тази релация $R1$ до релация $R2$.
- ▶ Например, ако релацията $R1(A1, A2, A3)$, то релацията $R1(A1, A2)$ е проектираната релация $R1$ без атрибута $A3$.
- ▶ Въпросът които ни интересува е кои ФЗ от S ще бъдат в сила за новата релация $R2$?
 - ▶ Това ще бъдат тези ФЗ, които логически следват от S
 - ▶ И които включват само атрибутите от $R2$

Проектиране на ФЗ - Пример

► Задача:

- Нека ни е дадена релацията $R1(A, B, C, D)$ има следните ФЗ
 - $A \rightarrow B$
 - $B \rightarrow C$
 - $C \rightarrow D$
- Искаме да проектираме релацията $R1$ до релацията $R2$, състояща се от атрибутите (A, C, D) , т.е. без атрибута B . Питаме се кои ФЗ от горните са в сила за $R2$

► Решение:

- Универсалният подход е да намерим покритията на всичките 8 подмножества на множеството от атрибут $\{A, C, D\}$:
 $\{ \}, \{A\}, \{C\}, \{D\}, \{AC\}, \{AD\}, \{CD\}, \{ACD\}$

Проектиране на ФЗ - Пример

- ▶ Първо започваме от покритието на единичните атрибути, ако е необходимо преминаваме към покритието на множествата от два атрибута и т.н.
- ▶ Можем да приложим следните опростявания:
 - ▶ Празното множество и множеството от всички атрибути не може да предизвика нетривиални ФЗ
 - ▶ Ако знаем че покритието на едно множество X включва всички атрибути тогава няма да открием нови ФЗ, за надмножествата на X
- ▶ Като приложим тези правила за опростяване, към горния пример: $\{\}$, $\{A\}$, $\{C\}$, $\{D\}$, $\{AC\}$, $\{AD\}$, $\{CD\}$, $\{ACD\}$, остава да намерим покритието на $\{A\}$, $\{C\}$, $\{D\}$, $\{AC\}$, $\{AD\}$, $\{CD\}$.

Проектиране на ФЗ - Пример

- ▶ Като вземем предвид ФЗ, които са в сила за релацията $R1(A, B, C, D)$:
 - ▶ $A \rightarrow B$
 - ▶ $B \rightarrow C$
 - ▶ $C \rightarrow D$
- ▶ Покритието на $\{A\} = \{A, B, C, D\}$, от тук следва че **$A \rightarrow C$** и **$A \rightarrow D$** са верни за новата релация $R2$. ФЗ: $A \rightarrow B$ е вярна за първата релация $R1$, но няма смисъл за втората релация $R2$
- ▶ Намираме покритието на $\{C\} = \{C, D\}$, от тук следва че **$C \rightarrow D$** е вярна за новата релация $R2$
- ▶ Намираме покритието на $\{D\} = \{D\}$, от тук няма какво да добавим.
- ▶ Понеже покритието получено за $\{A\}$ съдържа всички атрибути на множеството $R2$, няма смисъл да разглежда подмножества съдържащи атрибута A . Причината е, че каквато и ФЗ да открием тя вече ще била добавена от по-предните стъпки.
- ▶ Единственото надмножество което има смисъл да се разгледа е покритието на $\{C, D\} = \{C, D\}$. Но от него също не получаваме нови ФЗ.
- ▶ Така ФЗ, които са в сила за $R2$ са: **$A \rightarrow C$** , **$A \rightarrow D$** и **$C \rightarrow D$**