

Σχεσιακό Μοντέλο



- Η δομή των Σχεσιακών Βάσεων Δεδομένων
- Περιορισμοί ακεραιότητας
- Σχεσιακή Άλγεβρα
- Αλλαγές στη Βάση Δεδομένων
- Θεωρήσεις

Βασική πηγή διαφανειών: Silberschatz et al., "Database System Concepts", 4/e
Εργαστήριο Πληροφοριακών Συστημάτων, Παν/μιο Πειραιώς (<http://infolab.cs.unipi.gr/>)
έκδοση: Φεβ. 2010



- ➔
- Η δομή των Σχεσιακών Βάσεων Δεδομένων
 - Περιορισμοί ακεραιότητας
 - Σχεσιακή Άλγεβρα
 - Αλλαγές στη Βάση Δεδομένων
 - Θεωρήσεις

Παραδείγματα σχέσεων (πινάκων)



account-number	branch-name	balance
A-101	Downtown	500
A-102	Perryridge	400
A-201	Brighton	900
A-215	Mianus	
A-217	Brighton	
A-222	Redwood	
A-305	Round Hill	

customer-name	account-number
Hayes	A-102
Johnson	A-101
Johnson	A-201
Jones	A-217
Lindsay	A-222
Smith	A-215
Turner	A-305

customer-name	customer-street	customer-city
Adams	Spring	Pittsfield
Brooks	Senator	Brooklyn
Curry	North	Rye
Glenn	Sand Hill	Woodside
Green	Walnut	Stamford
Hayes	Main	Harrison
Johnson	Alma	Palo Alto
Jones	Main	Harrison
Lindsay	Park	Pittsfield
Smith	North	Rye
Turner	Putnam	Stamford
Williams	Nassau	Princeton

ΒΔ: [2] Σχεσιακό Μοντέλο

3

ΠΑ.ΠΕΙ. – Γιάννης Θεοδωρίδης

Βασική δομή (1)



- Ένα **πεδίο τιμών** (domain) D είναι ένα σύνολο τιμών
 $D = \{d_1, d_2, \dots, d_n\}$
- Ένα **χαρακτηριστικό** (attribute) A ονοματίζει μια (ενδιαφέρουσα) ιδιότητα σε μία Σχέση και παίρνει τιμές από το αντίστοιχο πεδίο τιμών $D(A)$.
 Συμβολισμός: **$R.A$** ή **$R[A]$** , όπου R είναι το όνομα της Σχέσης
- Μια **πλειάδα** (tuple) t μιας Σχέσης $R(A_1, A_2, \dots, A_n)$ είναι μια (διατεταγμένη) λίστα τιμών $t = \langle v_1, v_2, \dots, v_n \rangle$, όπου κάθε τιμή v_i είναι ένα στοιχείο του πεδίου $D(A_i)$.
- Παράδειγμα:
 ("A-101", "Downtown", 500)
 π.χ. η τιμή 500 (από το πεδίο τιμών των μη αρνητικών αριθμών) εκφράζει μια ιδιότητα ενός τραπεζικού καταθετικού λογαριασμού (το «υπόλοιπο»).

ΒΔ: [2] Σχεσιακό Μοντέλο

4

ΠΑ.ΠΕΙ. – Γιάννης Θεοδωρίδης

Βασική δομή (2)



- Δοθέντων $D_1, D_2, \dots, D_n$ μια **σχέση (relation)** r είναι ένα υποσύνολο του $D_1 \times D_2 \times \dots \times D_n$
Άρα, μια σχέση είναι ένα σύνολο **n-πλειάδων (n-tuples)** $(a_1, a_2, \dots, a_n)$ όπου κάθε $a_i \in D_i$
- Παράδειγμα: έστω
 $customer-name = \{Jones, Smith, Curry, Lindsay\}$
 $customer-street = \{Main, North, Park\}$
 $customer-city = \{Harrison, Rye, Pittsfield\}$
 τότε $r = \{$ (Jones, Main, Harrison),
 (Smith, North, Rye),
 (Curry, North, Rye),
 (Lindsay, Park, Pittsfield) $\}$
 είναι μια σχέση πάνω στο
 $customer-name \times customer-street \times customer-city$

Τύποι Χαρακτηριστικών



- Κάθε **χαρακτηριστικό (attribute)** μιας σχέσης έχει ένα διακριτό όνομα
- Οι τιμές των χαρακτηριστικών (σε κανονικές συνθήκες) πρέπει να είναι **ατομικές**, δηλαδή μη διαιρέσιμες σε μικρότερα τμήματα
 - Π.χ. Τα πλειότιμα χαρακτηριστικά ή τα σύνθετα χαρακτηριστικά δεν είναι ατομικά
- Η ειδική τιμή **null** αποτελεί μέλος κάθε πεδίου τιμών
 - Η τιμή null προκαλεί επιπλοκές στον ορισμό πολλών λειτουργιών πάνω στις βάσεις δεδομένων

Στιγμιότυπο Σχέσης



- Η τρέχουσα κατάσταση (**στιγμιότυπο σχέσης - relation instance**) μιας σχέσης προσδιορίζεται από έναν **πίνακα (table)**
- Ένα στοιχείο t της σχέσης r είναι μία **πλειάδα (tuple)**, που εκφράζεται με μια **γραμμή (row)** σε έναν πίνακα
 - Δεν υπάρχει διάταξη στις πλειάδες (οι πλειάδες μπορούν να αποθηκευτούν σε μια οποιαδήποτε σειρά)

<i>account-number</i>	<i>branch-name</i>	<i>balance</i>
A-101	Downtown	500
A-102	Perryridge	400
A-201	Brighton	900
A-215	Mianus	700
A-217	Brighton	750
A-222	Redwood	700
A-305	Round Hill	350

‘χαρακτηριστικά’
(ή ‘πεδία’ ή ‘στήλες’)

‘πλειάδες’
(ή ‘εγγραφές’
ή ‘γραμμές’)

Βάση Δεδομένων



- Μια **βάση δεδομένων (database)** είναι μια συλλογή από σχέσεις
- Η πληροφορία έχει σπάσει σε κομμάτια, με κάθε σχέση να κρατά ένα κομμάτι πληροφορίας

π.χ.: *account*: πληροφορία για τους λογαριασμούς
depositor: πληροφορία για τους καταθέτες
customer: πληροφορία για τους πελάτες

- Αν κρατούσαμε όλη την πληροφορία σε μία σχέση π.χ.
bank(account-number, balance, customer-name, ..)
 το αποτέλεσμα θα ήταν
 - επανάληψη πληροφορίας (π.χ. όταν δύο πελάτες έχουν κοινό λογαριασμό)
 - συχνή χρήση τιμών null (π.χ. για τους πελάτες χωρίς λογαριασμούς)
- Η **θεωρία κανονικοποίησης (*)** μελετά πώς να σχεδιάζουμε σχεσιακά σχήματα

(*) θα μελετηθεί σε επόμενη ενότητα

Κλειδιά

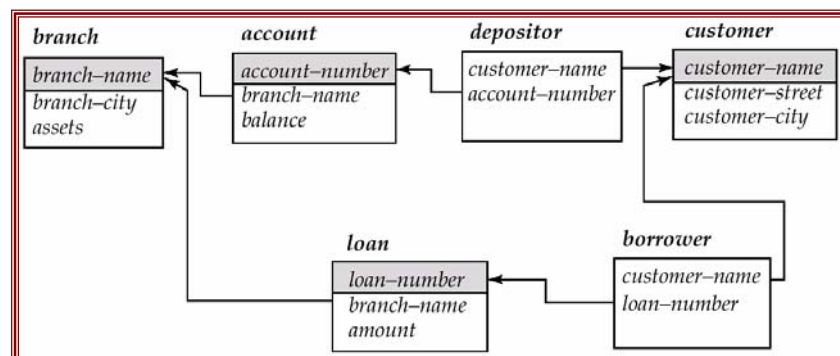


- Έστω $K \subseteq R$ ένα σύνολο χαρακτηριστικών ενός σχεσιακού σχήματος
- Το K είναι **υπερ-κλειδί (superkey)** του R εάν οι τιμές του K αρκούν για να ταυτοποιήσουν μια πλειάδα της οποίας σχέσης $r(R)$
 - Παράδειγμα: τόσο το $\{customer-name, customer-street\}$ όσο και το $\{customer-name\}$ αποτελούν και τα δύο υπερ-κλειδιά της *Customer*, αν υποθέσουμε ότι αποκλείεται δύο πελάτες να έχουν το ίδιο όνομα.
- Το K είναι **υποψήφιο κλειδί (candidate key)** εάν είναι ελάχιστο (minimal)
 - Παράδειγμα: το $\{customer-name\}$ είναι υποψήφιο κλειδί για την *Customer*, αφού είναι υπερ-κλειδί (με βάση τα παραπάνω) και επιπλέον κανένα υποσύνολό του δεν είναι υπερ-κλειδί.
- Ένα από τα υποψήφια κλειδιά αποφασίζεται (από το σχεδιαστή της βάσης δεδομένων) να είναι το **πρωτεύον κλειδί (primary key)** του R

Παράδειγμα ΒΔ (για μια τράπεζα)



- 6 πίνακες:
υποκαταστήματα (branches), πελάτες (customers),
λογαριασμοί (accounts), δάνεια (loans),
καταθέτες (depositors), δανειολήπτες (borrowers)





- Η δομή των Σχεσιακών Βάσεων Δεδομένων
- ➔ ▪ Περιορισμοί ακεραιότητας
- Σχεσιακή Άλγεβρα
- Αλλαγές στη Βάση Δεδομένων
- Θεωρήσεις

Περιορισμοί ακεραιότητας



- Οι **περιορισμοί ακεραιότητας** (*integrity constraints*) προσφέρουν προστασία από τυχόν απρόβλεπτους παράγοντες, εξασφαλίζοντας πως οι εξουσιοδοτημένες αλλαγές στη βάση δεδομένων δε θα οδηγήσουν σε απώλεια της **συνέπειας των δεδομένων** (*data consistency*).
 - **περιορισμοί πεδίου τιμών** (*domain constraints*): οι τιμές που εισάγονται στη βάση ελέγχονται για να διασφαλιστούν οι περιορισμοί πεδίου.
 - **ακεραιότητα οντότητας** (*entity integrity*): εξασφαλίζει σε ένα στιγμιότυπο σχέσης ότι θα μπορεί πάντα να γίνει ταυτοποίηση μιας οντότητας.
 - επιτυγχάνεται με την έννοια του πρωτεύοντος κλειδιού (*primary key*).
 - **αναφορική ακεραιότητα** (*referential integrity*): εξασφαλίζει ότι μια τιμή που εμφανίζεται σε ένα χαρακτηριστικό μιας σχέσης εμφανίζεται οπωσδήποτε και σε ένα χαρακτηριστικό κάποιας άλλης σχέσης.
 - επιτυγχάνεται με την έννοια του ξένου κλειδιού (*foreign key*).

Ακεραιότητα οντότητας



- **Ακεραιότητα οντότητας** (entity integrity): Το κύριο κλειδί PK στο σχήμα της σχέσης R δεν μπορεί να έχει κενές (null) ή επαναλαμβανόμενες τιμές σε πλειάδες μιας σχέσης $r(R)$.

$t[PK] \neq \text{NULL}$, για οποιαδήποτε t στην $r(R)$

$t_1[PK] \neq t_2[PK]$, για οποιεσδήποτε t_1, t_2 στην $r(R)$

- Παράδειγμα:
το χαρακτηριστικό
account-number είναι
πρωτεύον κλειδί της
σχέσης account

account-number	branch-name	balance
A-101	Downtown	500
A-215	Mianus	700
A-102	Perryridge	400
A-305	Round Hill	350
A-201	Brighton	900
A-222	Redwood	700
A-217	Brighton	750

Αναφορική ακεραιότητα



- **Αναφορική ακεραιότητα** (referential integrity): Έστω $r_1(R_1)$ και $r_2(R_2)$ είναι σχέσεις με πρωτεύοντα κλειδιά K_1 και K_2 αντίστοιχα.

- Το υποσύνολο α της R_2 είναι ξένο κλειδί (*foreign key*) που **αναφέρεται** στο K_1 της σχέσης r_1 , αν:

$$\forall t_2 \text{ στην } r_2 \exists t_1 \text{ στην } r_1 : t_1[K_1] = t_2[\alpha]$$

- Παράδειγμα:
το χαρακτηριστικό branch-name
είναι ξένο κλειδί της σχέσης
account και αναφέρεται στο
πρωτεύον κλειδί της σχέσης
branches (χαρακτηριστικό
branch-name)

account-number	branch-name	balance
A-101	Downtown	500
A-215	Mianus	700
A-102	Perryridge	400
A-305	Round Hill	350
A-201	Brighton	900
A-222	Redwood	700
A-217	Brighton	750

Έλεγχος αναφορικής ακεραιότητας κατά την τροποποίηση της βάσης



- **Εισαγωγή.** Αν μία πλειάδα t_2 εισαχθεί στην r_2 , το σύστημα πρέπει να εξασφαλίσει ότι υπάρχει μία πλειάδα t_1 στην r_1 τέτοια ώστε $t_1[K] = t_2[\alpha]$.
- **Διαγραφή.** Αν μία πλειάδα t_1 διαγραφεί από την r_1 , το σύστημα πρέπει να υπολογίσει το σύνολο των πλειάδων της r_2 που αναφέρονται στην πλειάδα t_1 . Αν το σύνολο αυτό δεν είναι κενό, θα πρέπει είτε να απορριφθεί η εντολή διαγραφής ως λάθος, είτε να διαγραφούν οι πλειάδες που αναφέρονται στην πλειάδα t_1 (είναι πιθανό να προκύψουν διαδοχικές διαγραφές).
- **Ενημέρωση.** Διακρίνουμε δύο περιπτώσεις:
 - Αν ενημερωθεί μία πλειάδα t_2 της r_2 και η ενημέρωση αυτή αλλάξει την τιμή του ξένου κλειδιού α , τότε βλ. Εισαγωγή.
 - Αν ενημερωθεί μία πλειάδα t_1 της r_1 και η ενημέρωση αλλάξει την τιμή του πρωτεύοντος κλειδιού K , τότε βλ. Διαγραφή.

- Η δομή των Σχεσιακών Βάσεων Δεδομένων
- Περιορισμοί ακεραιότητας
- ➔ ▪ Σχεσιακή Άλγεβρα
- Αλλαγές στη Βάση Δεδομένων
- Θεωρήσεις



Γλώσσες Ερωτημάτων



- Μια **γλώσσα ερωτημάτων (query language)** είναι μια γλώσσα με την οποία ο χρήστης ζητά πληροφορίες από τη βάση δεδομένων.
- Κατηγορίες γλωσσών ερωτημάτων
 - διαδικαστικές vs. μη-διαδικαστικές
 - Παράδειγμα διαδικαστικής γλώσσας: Σχεσιακή Άλγεβρα (Relational Algebra)
 - Παράδειγμα μη-διαδικαστικής γλώσσας: SQL

Σχεσιακή Άλγεβρα



- Διαδικαστική γλώσσα
- 6 βασικοί τελεστές ...
 - Επιλογή (select)
 - Προβολή (project)
 - Ένωση (union)
 - Διαφορά συνόλων (set difference)
 - Καρτεσιανό γινόμενο (Cartesian product)
 - Μετονομασία (rename)
- ... και άλλοι παραγόμενοι από τους 6 βασικούς
 - Τομή συνόλων (set intersection)
 - σύνδεση (join) και παραλλαγές της
 - Διαίρεση (division)
 - Εκχώρηση (assignment)
- Οι τελεστές λαμβάνουν ως εισόδους δύο ή περισσότερες σχέσεις και δίνουν ως έξοδο μια νέα σχέση (το αποτέλεσμα του τελεστή)

Επιλογή (Select)



Σχέση r

A	B	C	D
α	x	1	7
β	y	5	3
γ	y	12	3
δ	z	23	10

$\sigma_{B="y" \wedge D > 2}(r)$

A	B	C	D
β	y	5	3
γ	y	12	3

Ορισμός:

$$\sigma_p(r) = \{t \mid t \in r \text{ and } p(t)\}$$

όπου r είναι όνομα σχέσης και p είναι μια λογική συνθήκη που αποτελείται από **όρους** συνδεδεμένους με: $\wedge$ (**and**), $\vee$ (**or**), $\neg$ (**not**)

Κάθε **όρος** είναι ένα από:

<attribute> op <attribute> ή
<constant>

όπου op είναι ένας τελεστής ($=, \neq, >, \geq, <, \leq$)

Προβολή (Project)



Σχέση r

A	B	C	D
α	x	1	7
β	y	5	3
γ	y	12	3
δ	z	23	10

$\Pi_{B,D}(r)$

B	D
x	7
y	3
y	3
z	10

Συμβολισμός:

$$\Pi_{A_1, A_2, \dots, A_k}(r)$$

όπου r είναι όνομα σχέσης και $A_1, A_2, \dots, A_k$ είναι ονόματα χαρακτηριστικών

Το αποτέλεσμα είναι μια σχέση με k στήλες που προκύπτει αν διαγράψουμε τις στήλες που δεν εμφανίζονται στη λίστα χαρακτηριστικών

Οι διπλοεγγραφές διαγράφονται

- θυμηθείτε ότι οι σχέσεις είναι σύνολα

Ένωση (Union)



Σχέσεις r, s :

A	B	C	D
α	x	1	7
β	y	5	3
γ	y	12	3
δ	z	23	10

A	B	C	D
ϵ	x	8	3
δ	z	23	10

A	B	C	D
α	x	1	7
β	y	5	3
γ	y	12	3
δ	z	23	10
ϵ	x	8	3

Σχέση $r \cup s$:

- Συμβολισμός: $r \cup s$
- Ορισμός:

$$r \cup s = \{t \mid t \in r \vee t \in s\}$$
- Για να είναι έγκυρη η ένωση $r \cup s$ πρέπει οι σχέσεις r, s να είναι **συμβατές**:
 - Οι σχέσεις r, s να έχουν τον ίδιο **βαθμό** (ίδιο αριθμό χαρακτηριστικών)
 - Τα πεδία ορισμού των αντίστοιχων χαρακτηριστικών να είναι **συμβατά** (π.χ. η 2^η στήλη της r να έχει τον ίδιο τύπο τιμών με τη 2^η στήλη της s)

ΒΔ: [2] Σχεσιακό Μοντέλο

21

ΠΑ.ΠΕΙ. – Γιάννης Θεοδωρίδης

Διαφορά Συνόλων (Set Difference)



Σχέσεις r, s :

A	B	C	D
α	x	1	7
β	y	5	3
γ	y	12	3
δ	z	23	10

A	B	C	D
ϵ	x	8	3
δ	z	23	10

Σχέση $r - s$:

A	B	C	D
α	x	1	7
β	y	5	3
γ	y	12	3

- Συμβολισμός $r - s$
- Ορισμός: $r - s = \{t \mid t \in r \wedge t \notin s\}$
- ισχύει μόνο μεταξύ συμβατών σχέσεων

ΒΔ: [2] Σχεσιακό Μοντέλο

22

ΠΑ.ΠΕΙ. – Γιάννης Θεοδωρίδης

Καρτεσιανό γινόμενο (Cartesian Product)



Σχέσεις r, s :

A	B	C	D
α	x	1	7
β	y	5	3
γ	y	12	3
δ	z	23	10

E	F
x	1
y	2

Σχέση $r \times s$:

A	B	C	D	E	F
α	x	1	7	x	1
α	x	1	7	y	2
β	y	5	3	x	1
β	y	5	3	y	2
γ	y	12	3	x	1
γ	y	12	3	y	2
δ	z	23	10	x	1
δ	z	23	10	y	2

- Συμβολισμός $r \times s$
- Ορισμός:

$$r \times s = \{t \mid t \in r \wedge q \in s\}$$
- Κάνουμε την παραδοχή ότι τα χαρακτηριστικά των $r(R)$ και $s(S)$ είναι ξένα μεταξύ τους (δηλαδή, $R \cap S = \emptyset$)
 - Αν δεν ισχύει αυτό, πρέπει να γίνουν μετονομασίες χαρακτηριστικών

Σύνθεση βασικών λειτουργιών



- Μπορούμε να κτίσουμε εκφράσεις σχεσιακής άλγεβρας που εμπλέκουν πολλαπλές λειτουργίες
- Παράδειγμα: $\sigma_{A=C}(r \times s)$

$r \times s$

$\sigma_{B="y" \wedge D > 2}(r \times s)$

A	B	C	D	E	F
β	y	5	3	x	1
β	y	5	3	y	2
γ	y	12	3	x	1
γ	y	12	3	y	2

A	B	C	D	E	F
α	x	1	7	x	1
α	x	1	7	y	2
β	y	5	3	x	1
β	y	5	3	y	2
γ	y	12	3	x	1
γ	y	12	3	y	2
δ	z	23	10	x	1
δ	z	23	10	y	2

Μετονομασία (Rename)



- Μας επιτρέπει να δίνουμε ονόματα, και άρα να αναφερόμαστε, στα αποτελέσματα των εκφράσεων σχεσιακής άλγεβρας
- Μας επιτρέπει να αναφερόμαστε σε μια σχέση με περισσότερα από ένα ονόματα

Παράδειγμα:

$$\rho_X(E)$$

επιστρέφει την έκφραση E με το όνομα X

Αν η έκφραση σχεσιακής άλγεβρας E έχει βαθμό n , τότε η έκφραση

$$\rho_X(A_1, A_2, \dots, A_n)(E)$$

επιστρέφει το αποτέλεσμα της E με το όνομα X και με τα χαρακτηριστικά μετονομασμένα σε $A_1, A_2, \dots, A_n$

Τομή συνόλων (Set Intersection)



Σχέσεις r, s :

A	B	C	D
α	x	1	7
β	y	5	3
γ	y	12	3
δ	z	23	10

A	B	C	D
ϵ	x	8	3
δ	z	23	10

- Συμβολισμός: $r \cap s$
- Ορισμός:
 $r \cap s = \{t \mid t \in r \wedge t \in s\}$
- ισχύει μόνο μεταξύ συμβατών σχέσεων

$$r \cap s = r - (r - s)$$

Σχέση $r \cap s$:

A	B	C	D
δ	z	23	10

Σύνδεση (Join)



Σχέσεις r, s :

A	B	C	D
α	x	1	7
β	y	5	3
γ	y	12	3
δ	z	23	10

E	F
x	1
y	2
w	5

- Συμβολισμός: $r \bowtie_p s$
- Έστω r και s σχέσεις και p μια λογική συνθήκη. Τότε, $r \bowtie_p s$ είναι ένα υποσύνολο της σχέσης $r \times s$ που περιλαμβάνει μόνο τις πλειάδες που ικανοποιούν τη συνθήκη p

Σχέση $r \bowtie_{C=F} s$:

A	B	C	D	E	F
α	x	1	7	x	1
β	y	5	3	w	5

$$r \bowtie_p s = \sigma_p (r \times s)$$

ΒΔ: [2] Σχεσιακό Μοντέλο

27

ΠΑ.ΠΕΙ. – Γιάννης Θεοδωρίδης

Διαίρεση (Division)



Σχέσεις r, s :

A	B
α	1
α	2
α	3
β	1
γ	1
δ	1
δ	3
δ	4
ϵ	6
ϵ	1
β	2

B
1
2

$r \div s$:

A
α
β

- Συμβολισμός: $r \div s$
- Κατάλληλο για ερωτήσεις που περιλαμβάνουν τη φράση «για όλα»
- Έστω r και s σχέσεις πάνω στα σχήματα R και S αντίστοιχα, όπου
 - $R = (A_1, \dots, A_m, B_1, \dots, B_n)$
 - $S = (B_1, \dots, B_n)$
 το αποτέλεσμα της $r \div s$ είναι μια σχέση πάνω στο σχήμα $R - S = (A_1, \dots, A_m)$

- Ορισμός:

$$r \div s = \{ t \mid t \in \Pi_{R-S}(r) \wedge \forall u \in s (tu \in r) \}$$

$$r \div s = \Pi_{R-S}(r) - \Pi_{R-S}((\Pi_{R-S}(r) \times s) - \Pi_{R-S,S}(r))$$

ΒΔ: [2] Σχεσιακό Μοντέλο

28

ΠΑ.ΠΕΙ. – Γιάννης Θεοδωρίδης

Εκχώρηση (Assignment)



- Η εκχώρηση ($\leftarrow$) παρέχει ένα πρακτικό τρόπο να εκφράσουμε σύνθετα ερωτήματα
 - ένα σύνθετο ερώτημα είναι μια σειρά εκχωρήσεων και εκφράσεων
 - εκχωρήσεις γίνονται μόνο σε ονόματα προσωρινών σχέσεων (μεταβλητών)
- Παράδειγμα: Μπορούμε να γράψουμε τη διαίρεση $r \div s$ ως

$$\begin{aligned} temp1 &\leftarrow \Pi_{R-S}(r) \\ temp2 &\leftarrow \Pi_{R-S}((temp1 \times s) - \Pi_{R-S,S}(r)) \\ result &= temp1 - temp2 \end{aligned}$$

- Το αποτέλεσμα της έκφρασης δεξιά του $\leftarrow$ εκχωρείται στη σχέση-μεταβλητή που δηλώνεται αριστερά του $\leftarrow$

Συνάθροιση (Aggregation)



- Μια **συνάρτηση συνάθροισης** (aggregate function) λαμβάνει ως είσοδο μια συλλογή τιμών και επιστρέφει ως αποτέλεσμα μια μοναδική τιμή.
 - Παραδείγματα: **avg**: μέση τιμή, **min**: ελάχιστη τιμή, **max**: μέγιστη τιμή, **sum**: άθροισμα τιμών, **count**: πλήθος τιμών
- Μια **λειτουργία συνάθροισης** (aggregate operation) στη σχεσιακή άλγεβρα δηλώνεται ως:

$G_1, G_2, \dots, G_n \mathcal{G}_{F_1(A_1), F_2(A_2), \dots, F_n(A_n)}(E)$ όπου

- E είναι μια έκφραση σχεσιακής άλγεβρας
- $G_1, G_2, \dots, G_n$ είναι λίστα χαρακτηριστικών πάνω στα οποία θα κάνουμε ομαδοποίηση τιμών
- F_i είναι συναρτήσεις συνάθροισης
- A_i είναι ονόματα χαρακτηριστικών

Σχέση r

A	B	C	D
α	x	1	7
β	y	5	3
γ	y	12	3
δ	z	23	10

$\mathcal{G}_{sum(C)}(r)$

B	sum-C
x	1
y	17
z	23

$\mathcal{G}_{sum(C)}(r)$

sum-C

41

Παραλλαγές της σύνδεσης



- Υπάρχουν πολλές μορφές σύνδεσης – όλες συνδυάζουν δύο Σχέσεις για την δημιουργία μιας τρίτης
 - φυσική σύνδεση (natural join)
 - ημι-σύνδεση (semi-join)
 - εξωτερική σύνδεση (outer join)

Φυσική σύνδεση (Natural Join)



Σχέσεις r, s :

A	B	C	D
α	x	1	7
β	y	5	3
γ	y	12	3
δ	z	23	10

B	E
x	1
Y	2
w	5

Σχέση $r \bowtie s$:

A	B	C	D	E
α	x	1	7	1
β	y	5	3	2
γ	y	12	3	2

- Συμβολισμός: $r \bowtie s$
- Έστω $r(R)$ και $s(S)$ σχέσεις.
 $r \bowtie s$ είναι μια σχέση πάνω στο σχήμα $R \times S$ που προκύπτει ως εξής:
 - Έστω δύο πλειάδες,
 $t_r \in r$ και $t_s \in s$
 - Εάν οι t_r και t_s έχουν ίδια τιμή για καθένα από τα χαρακτηριστικά του $R \cap S$, προσθέτουμε μια πλειάδα t στο αποτέλεσμα, όπου
 - t έχει ίδια τιμή με t_r στην r
 - t έχει ίδια τιμή με t_s στην s

$$r \bowtie s = \Pi_{r.A, r.B, r.C, r.D, s.E} (\sigma_{r.B = s.B} (r \times s))$$

Ημι-σύνδεση (Semi-Join)



- Παραλλαγή της λειτουργίας σύνδεσης στην περίπτωση που μας ενδιαφέρει η πληροφορία μιας μόνο σχέσης

Συμβολισμός: $r \bowtie_p s$

- κρατά τα χαρακτηριστικά μόνο της αριστερής σχέσης
- ισοδύναμη με μια σύνδεση ακολουθούμενη από μια προβολή.

$$r \bowtie_p s = \pi_L (r \bowtie s) \mid L: \text{τα χαρακτηριστικά της } r$$

Σχέσεις r, s :

A	B	C	D
α	x	1	7
β	y	5	3
γ	y	12	3
δ	z	23	10

E	F
x	1
Y	2
w	5

Σχέση: $r \bowtie_{C=F} s$

A	B	C	D
α	x	1	7
β	y	5	3

Εξωτερική σύνδεση (Outer Join)



- Επέκταση της λειτουργίας σύνδεσης για την αποφυγή απώλειας πληροφορίας
- Υπολογίζει τη σύνδεση και κατόπιν προσθέτει στο αποτέλεσμα της σύνδεσης πλειάδες της μιας σχέσης που δεν ταιριάζουν με πλειάδες της άλλης σχέσης
- Κάνει χρήση τιμών null
 - Η τιμή null δηλώνει είτε ότι η τιμή είναι άγνωστη είτε ότι δεν υπάρχει
 - Οποιαδήποτε σύγκριση με τιμή null είναι ψευδής (false) εξ' ορισμού
- Παραλλαγές:
 - Εξωτερική από αριστερά σύνδεση (left-outer join)
 - Εξωτερική από δεξιά σύνδεση (right-outer join)
 - Πλήρης εξωτερική σύνδεση (full-outer join)

Παραδείγματα εξωτερικής σύνδεσης



Σχέσεις r, s:

A	B	C	D
α	x	1	7
β	y	5	3
γ	y	12	3
δ	z	23	10

B	E
x	1
γ	2
w	5

left-outer join $r \bowtie_l s$

A	B	C	D	E
α	x	1	7	1
β	y	5	3	2
γ	y	12	3	2
δ	z	23	10	

right-outer join $r \bowtie_r s$

A	B	C	D	E
α	x	1	7	1
β	y	5	3	2
γ	y	12	3	2
	w			5

full-outer join
 $r \bowtie_{fs} s$

A	B	C	D	E
α	x	1	7	1
β	y	5	3	2
γ	y	12	3	2
δ	z	23	10	
	w			5

ΒΔ: [2] Σχεσιακό Μοντέλο

35

ΠΑ.ΠΕΙ. – Γιάννης Θεοδωρίδης

Παράδειγμα Τράπεζας



branch (branch-name, branch-city, assets)

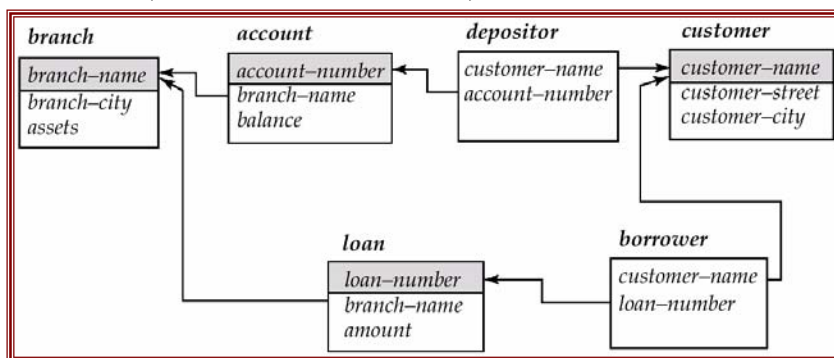
customer (customer-name, customer-street, customer-city)

account (account-number, branch-name, balance)

loan (loan-number, branch-name, amount)

depositor (customer-name, account-number)

borrower (customer-name, loan-number)



ΒΔ: [2] Σχεσιακό Μοντέλο

36

ΠΑ.ΠΕΙ. – Γιάννης Θεοδωρίδης

Παραδείγματα ερωτήσεων (1)



- «Βρες τα δάνεια ποσού άνω των 1200»

$\sigma_{amount > 1200} (loan)$

- «Βρες τα δάνεια ποσού άνω των 1200 (εμφάνιση των αριθμών δανείων μόνο)»

$\Pi_{loan-number} (\sigma_{amount > 1200} (loan))$

account-number	branch-name	balance
A-101	Downtown	500
A-215	Mianus	700
A-102	Perryridge	400
A-305	Round Hill	350
A-201	Brighton	900
A-222	Redwood	700
A-217	Brighton	750

Παραδείγματα ερωτήσεων (2)



- «Βρες τα ονόματα των πελατών που έχουν λογαριασμό ή δάνειο»

$\Pi_{customer-name} (borrower) \cup \Pi_{customer-name} (depositor)$

- «Βρες τα ονόματα των πελατών που έχουν λογαριασμό και δάνειο»

customer-name	account-number
Hayes	A-102
Johnson	A-101
Johnson	A-201
Jones	A-217
Lindsay	A-222
Smith	A-215
Turner	A-305

$\Pi_{customer-name} (borrower) \cap \Pi_{customer-name} (depositor)$

Παραδείγματα ερωτήσεων (3)



- «Βρες τα ονόματα των πελατών που έχουν καταθετικό λογαριασμό στο υποκατάστημα του Perryridge»

account-number	branch-name	balance
A-101	Downtown	500
A-102	Perryridge	400
		900
		700
customer-name	account-number	
Hayes	A-102	750
Johnson	A-101	700
Johnson	A-201	350
Jones	A-217	
Lindsay	A-222	
Smith	A-215	
Turner	A-305	

- 1ος τρόπος (με σύνδεση)

$$\Pi_{\text{customer-name}}(\sigma_{\text{branch-name} = \text{"Perryridge"}}(\text{depositor} \bowtie \text{account}))$$

- 2ος τρόπος (με καρτεσιανό γινόμενο)

$$\Pi_{\text{customer-name}}(\sigma_{\text{account.account-number} = \text{depositor.account-number}}(\sigma_{\text{branch-name} = \text{"Perryridge"}}(\text{account})) \times \text{depositor})$$

Παραδείγματα ερωτήσεων (4)



- «Βρες το λογαριασμό με το μεγαλύτερο υπόλοιπο» (χωρίς χρήση της max)

$$\Pi_{\text{account-number, balance}}(\text{account}) - \Pi_{d.\text{account-number, d.balance}}(\sigma_{\text{account.balance} > d.\text{balance}}(\text{account} \times \rho_d(\text{account})))$$

account-number	branch-name	balance
A-101	Downtown	500
A-102	Perryridge	400
		900
		700
customer-name	account-number	
Hayes	A-102	750
Johnson	A-101	700
Johnson	A-201	350
Jones	A-217	
Lindsay	A-222	
Smith	A-215	
Turner	A-305	

- «Βρες τους πελάτες που έχουν λογαριασμούς σε όλα τα υποκαταστήματα της πόλης Brooklyn»

$$\Pi_{\text{customer-name, branch-name}}(\text{depositor} \bowtie \text{account}) \div \Pi_{\text{branch-name}}(\sigma_{\text{branch-city} = \text{"Brooklyn"}}(\text{branch}))$$



- Η δομή των Σχεσιακών Βάσεων Δεδομένων
- Περιορισμοί ακεραιότητας
- Σχεσιακή Άλγεβρα
- ➔ ▪ Αλλαγές στη Βάση Δεδομένων
- Θεωρήσεις

Αλλαγές στη Βάση Δεδομένων



- Το περιεχόμενο της ΒΔ μπορεί να αλλάξει με χρήση των ακόλουθων λειτουργιών:
 - Διαγραφή (Deletion)
 - Εισαγωγή (Insertion)
 - Τροποποίηση (Updating)
- Όλες αυτές οι λειτουργίες εκφράζονται με χρήση του τελεστή εκχώρησης:
 - Διαγραφή πλειάδων: $r \leftarrow r - E$
 - Εισαγωγή πλειάδων: $r \leftarrow r \cup E$
 - Τροποποίηση πλειάδων: $r \leftarrow \Pi_{F_1, F_2, \dots, F_n}(r)$

Διαγραφή πλειάδων



- Ένα αίτημα διαγραφής εκφράζεται όμοια με ένα αίτημα ερώτησης, απλά αντί να εμφανίζονται οι πλειάδες του αποτελέσματος στο χρήστη, οι πλειάδες αυτές διαγράφονται από τη βάση δεδομένων
 - Διαγράφουμε ολόκληρες πλειάδες και όχι τιμές συγκεκριμένων γνωρισμάτων
- Η διαγραφή εκφράζεται σε σχεσιακή άλγεβρα ως:

$$r \leftarrow r - E$$

όπου r είναι μια σχέση και E είναι ερώτηση σχεσιακής άλγεβρας

- Παραδείγματα:
 - Να διαγραφούν όλοι οι λογαριασμοί του υποκαταστήματος Perryridge

$$account \leftarrow account - \sigma_{branch-name = "Perryridge"}(account)$$

Εισαγωγή πλειάδων



- Για να εισάγουμε δεδομένα σε μια σχέση, κάνουμε ένα από τα εξής:
 - προδιαγράφουμε την πλειάδα που θα εισαχθεί, είτε
 - γράφουμε μια έκφραση, το αποτέλεσμα της οποίας αποτελεί το σύνολο πλειάδων που θα εισαχθεί
- Η εισαγωγή εκφράζεται σε σχεσιακή άλγεβρα ως:

$$r \leftarrow r \cup E$$

όπου r είναι μια σχέση και E είναι ερώτηση σχεσιακής άλγεβρας

- Η εισαγωγή μίας μόνο πλειάδας εκφράζεται με την E να είναι μια σχέση που περιέχει μια πλειάδα
- Παράδειγμα: «Να εισαχθεί στη ΒΔ η πληροφορία ότι ο πελάτης Smith άνοιξε στο υποκατάστημα Perryridge ένα λογαριασμό με κωδικό A-973 και ποσό 1200»

$$account \leftarrow account \cup \{("A-973", "Perryridge", 1200)\}$$

$$depositor \leftarrow depositor \cup \{("Smith", "A-973")\}$$

Τροποποίηση πλειάδων



- Η τροποποίηση αποτελεί ένα μηχανισμό για να αλλάξουμε την τιμή ενός χαρακτηριστικού σε μια πλειάδα χωρίς να αλλάξουμε όλες τις τιμές της πλειάδας
- Για να το πετύχουμε αυτό, χρησιμοποιούμε τον τελεστή (γενικευμένης) προβολής

$$r \leftarrow \prod_{F_1, F_2, \dots, F_n} (r)$$

όπου F_j είναι

- το i -οστό χαρακτηριστικό της r , αν αυτό δεν τροποποιείται, είτε
- (αν το συγκεκριμένο χαρακτηριστικό είναι να τροποποιηθεί) μια έκφραση που εμπλέκει μόνο σταθερές και χαρακτηριστικά της r , δίνοντας έτσι νέα τιμή στο συγκεκριμένο χαρακτηριστικό
- Παράδειγμα: «Να αυξηθούν κατά 5% τα υπόλοιπα όλων των λογαριασμών»

$$account \leftarrow \prod_{account-number, branch-name, balance * 1.05} (account)$$

Άλλα παραδείγματα



- «Δώρισε σε όλους τους δανειολήπτες του υποκαταστήματος Perryridge έναν καταθετικό λογαριασμό με αρχικό ποσό 200. Ο κωδικός του λογαριασμού να οριστεί από τον κωδικό του αντίστοιχου δανείου»

$$r_1 \leftarrow (\sigma_{branch-name = "Perryridge"} (borrower \bowtie loan))$$

$$account \leftarrow account \cup \prod_{loan-number, branch-name, 200} (r_1)$$

$$depositor \leftarrow depositor \cup \prod_{customer-name, loan-number} (r_1)$$

- «Να δοθεί αύξηση 5% σε όλους τους λογαριασμούς. Ειδικά για τους λογαριασμούς άνω των 10.000 η αύξηση να είναι 6%»

$$account \leftarrow \prod_{AN, BN, BAL * 1.06} (\sigma_{BAL > 10000} (account)) \cup \prod_{AN, BN, BAL * 1.05} (\sigma_{BAL \leq 10000} (account))$$



- Η δομή των Σχεσιακών Βάσεων Δεδομένων
- Περιορισμοί ακεραιότητας
- Σχεσιακή Άλγεβρα
- Αλλαγές στη Βάση Δεδομένων
- ➔ ▪ Θεωρήσεις

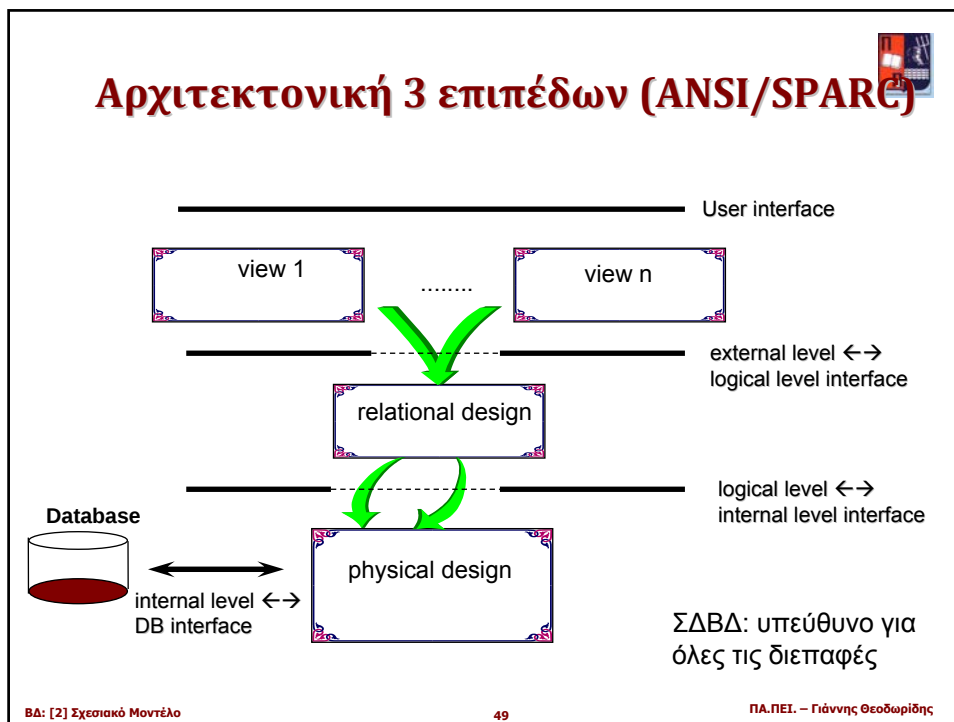
Θεωρήσεις (ή Όψεις)



- Σε ορισμένες περιπτώσεις, δεν επιθυμούμε να έχουν όλοι οι χρήστες το δικαίωμα πρόσβασης σε όλη τη βάση δεδομένων
- Παράδειγμα: σε κάποιον δίνεται το δικαίωμα πρόσβασης σε όλη την πληροφορία για τα δάνεια εκτός του ποσού του δανείου. Αυτός ο χρήστης πρέπει να 'βλέπει' μια σχέση που περιγράφεται σε σχεσιακή άλγεβρα ως εξής:

$$\Pi_{\text{customer-name, loan-number}} (\text{borrower} \bowtie \text{loan})$$

- Μια «εικονική» σχέση που δεν αποτελεί μέρος του εννοιολογικού μοντέλου αλλά είναι προσπελάσιμη από τους χρήστες ονομάζεται **θεώρηση ή όψη (view)**.



Ορισμός Θεώρησης

- Μια θεώρηση ορίζεται με την εντολή **create view** που έχει τη μορφή

create view v as <query expression>

 όπου <query expression> είναι μια έγκυρη έκφραση ερωτήματος σχεσιακής άλγεβρας και v είναι το όνομα της θεώρησης.
- Από τη στιγμή που έχει οριστεί η θεώρηση, μπορεί να χρησιμοποιηθεί σαν οποιαδήποτε άλλη σχέση.
- Ο ορισμός μιας θεώρησης δεν έχει την ίδια βαρύτητα με τον ορισμό μιας σχέσης (με αποτίμηση της query expression)
 - Ουσιαστικά, ο ορισμός μιας θεώρησης είναι κάτι σαν αποθήκευση με ένα όνομα της query expression ώστε να χρησιμοποιηθεί στη συνέχεια σε άλλα ερωτήματα.

Παραδείγματα Θεωρήσεων



- Έστω μια θεώρηση (με όνομα *all-customer*) που αποτελείται από τα υποκαταστήματα και τους πελάτες του κάθε υποκαταστήματος (είτε καταθέτες είτε δανειολήπτες)

create view *all-customer* as

$$\Pi_{branch-name, customer-name} (depositor \bowtie account) \\ \cup \Pi_{branch-name, customer-name} (borrower \bowtie loan)$$

- Παράδειγμα: «Να βρεθούν οι πελάτες του υποκαταστήματος Perryridge»

$$\Pi_{customer-name} \\ (\sigma_{branch-name = \text{"Perryridge"}} (all-customer))$$

Αλλαγές στη ΒΔ μέσω Θεωρήσεων (1)



- Εκφράσεις σχεσιακής άλγεβρας που αφορούν σε αλλαγές στη ΒΔ (εισαγωγές / διαγραφές / τροποποιήσεις) και περιλαμβάνουν θεωρήσεις μεταφράζονται σε εκφράσεις που περιλαμβάνουν τις βασικές σχέσεις της ΒΔ.
- Έστω η θεώρηση *branch-loan* με όλες τις πληροφορίες για τα δάνεια εκτός του ποσού:

create view *branch-loan* as

$$\Pi_{branch-name, loan-number} (loan)$$

- Αφού επιτρέπεται μια θεώρηση να χρησιμοποιείται όπως μια (βασική) σχέση, μια τέτοια έκφραση είναι έγκυρη:

$$branch-loan \leftarrow branch-loan \cup \{(\text{"Perryridge"}, \text{"L-37"})\}$$

Αλλαγές στη ΒΔ μέσω Θεωρήσεων (2)



- Η προηγούμενη εντολή εισαγωγής πλειάδας μεταφράζεται σε έκφραση που περιέχει τη βασική σχέση loan από την οποία προέκυψε η θεώρηση branch-loan.
- Πρόβλημα: για την εισαγωγή μιας πλειάδας στη σχέση loan χρειάζεται μια τιμή για το χαρακτηριστικό amount.
- Δύο εναλλακτικές λύσεις.
 - η εισαγωγή απορρίπτεται με κατάλληλο μήνυμα σφάλματος ή
 - εισάγεται στη σχέση loan η πλειάδα ("L-37", "Perryridge", null)

Αλλαγές στη ΒΔ μέσω Θεωρήσεων (3)



- Μερικές φορές είναι αδύνατες οι αλλαγές μέσω θεωρήσεων.
- Παράδειγμα:
 - create view v as σbranch-name = "Perryridge" (account))
 - $v \leftarrow v \cup (L-99, \text{Downtown}, 23)$
- Άλλες φορές η μετάφραση που απαιτείται να γίνει δεν μπορεί να προσδιοριστεί ακριβώς.
- Παράδειγμα:
 - $\text{all-customer} \leftarrow \text{all-customer} \cup \{("Perryridge", "John")\}$
 - Πρέπει να επιλεγεί είτε η σχέση loan είτε η σχέση account και να γίνει εκεί εισαγωγή μιας πλειάδας (με κάποιο νέο κωδικό!)

Εξάρτηση Θεώρησης από άλλη



- Στην έκφραση που ορίζει μια θεώρηση μπορεί να εμπλέκεται μια άλλη θεώρηση
- Η θεώρηση v_1 λέμε ότι **εξαρτάται άμεσα** από τη θεώρηση v_2 εάν η v_2 εμπλέκεται στην έκφραση με την οποία ορίζεται η v_1
- Η θεώρηση v_1 λέμε ότι **εξαρτάται** από τη θεώρηση v_2 εάν είτε η v_1 εξαρτάται άμεσα από τη v_2 είτε υπάρχει ένα μονοπάτι εξαρτήσεων από τη v_1 στη v_2
- Η θεώρηση v λέμε ότι είναι αναδρομική εάν εξαρτάται από τον εαυτό της.
 - Οι αναδρομικές θεωρήσεις μπορεί να είναι προβληματικές στο χειρισμό τους