

Κεφάλαιο 3

Σχεσιακό Μοντέλο

Στο κεφάλαιο αυτό παρουσιάζεται το σχεσιακό μοντέλο βάσεων δεδομένων, και αναλύονται τα δομικά του χαρακτηριστικά, οι βασικές του ιδιότητες, και ο τρόπος με τον οποίο μπορεί να προκύψει από το μοντέλο οντοτήτων συσχετίσεων.

Το σχεσιακό μοντέλο αναπαράστασης των δεδομένων μιας εφαρμογής (**relational data model**) καθιερώθηκε από τον Codd το 1970 και αποτέλεσε ένα από τα πιο απλά και ευέλικτα μοντέλα αυτού του είδους. Σε αυτό το μοντέλο, τα δεδομένα μιας εφαρμογής αναπαρίστανται ως ένα σύνολο από **σχέσεις (relations)** οι οποίες μπορεί να είναι **πίνακες** ή **αρχεία**. Στις πιο πολλές περιπτώσεις υιοθετείται η χρήση **πινάκων (tables)** που περιέχουν ένα πλήθος **γραμμών (rows)** και **στηλών (columns)**. Η κάθε μια από αυτές τις γραμμές – οι οποίες στην ορολογία του μοντέλου ονομάζονται και **πλειάδες (tuples)** – περιέχει ένα σύνολο **απλών πεδίων (attributes)** τα οποία συσχετίζονται μεταξύ τους. Επειδή όπως θα δούμε στις επόμενες παραγράφους οι πίνακες χρησιμοποιούνται για την αναπαράσταση των τύπων οντοτήτων καθώς και των τύπων συσχετίσεων που υφίστανται ανάμεσά τους, μπορούμε να θεωρήσουμε κάθε μια από τις γραμμές ενός πίνακα σαν ένα **στιγμιότυπο οντότητας ή συσχέτισης** ανάλογα με το αντικείμενο στο οποίο αναφέρεται. Χαρακτηριστικό παράδειγμα ενός πίνακα όπως αυτός ορίζεται από το σχεσιακό μοντέλο, παρουσιάζεται στη συνέχεια.

STUDENT						
Name	SSN	Home Phone	Address	Office Phone	Age	GPA
Bayer	305612435	3731616	2918 Bluebonnet Lane	NULL	19	3.21
Ashly	381621245	3754409	125 Kirby Road	NULL	18	2.89
David	422112320	NULL	3452 Elgin Road	7491253	25	3.53
Cooper	489221100	3769821	265 Lark Lane	7496492	28	3.93
Benson	533691238	8398461	7384 Fontana Lane	NULL	19	3.25

Πίνακας 2 : Παράδειγμα πίνακα στο σχεσιακό μοντέλο αναπαράστασης δεδομένων

Στον παραπάνω πίνακα η κάθε **πλειάδα** ή **εγγραφή (record)** αναφέρεται και σε ένα ξεχωριστό στιγμιότυπο του τύπου οντότητας **STUDENT**. Η κάθε μια από αυτές τις εγγραφές αποτελείται από ένα σύνολο χαρακτηριστικών ή πεδίων τα οποία

λαμβάνουν συγκεκριμένες τιμές, διαφορετικές εν γένει για κάθε εγγραφή. Έτσι, ο πίνακας **STUDENT** περιέχει τα πεδία **Name**, **SSN**, **Home Phone**, **Address**, **Office Phone**, **Age** και **GPA**. Επειδή οι τιμές της κάθε στήλης αναφέρονται ουσιαστικά στο ίδιο πεδίο του πίνακα, είναι προφανές πως θα πρέπει να ανήκουν στον ίδιο τύπο δεδομένων.

ΘΕΜΕΛΙΩΔΕΙΣ ΟΡΙΣΜΟΙ ΤΟΥ ΣΧΕΣΙΑΚΟΥ ΜΟΝΤΕΛΟΥ

Κάθε **πεδίο** ή **στήλη** ενός πίνακα, δέχεται τιμές οι οποίες ανήκουν σε ένα συγκεκριμένο και καθορισμένο εκ των προτέρων **σύνολο τιμών (domain)**. Το είδος των τιμών αυτού του συνόλου καθορίζεται **από τον τύπο δεδομένων του πεδίου του πίνακα**, ο οποίος με τη σειρά του ορίζεται κατά το στάδιο της λογικής σχεδίασης της εφαρμογής. Έτσι, ένα πεδίο που εκφράζει την ηλικία, θα αναπαρίσταται από μια ακέραια ποσότητα. Όσον αφορά το σύνολο των επιτρεπτών τιμών για αυτό το πεδίο, αυτό εξαρτάται και πάλι από τη φύση του προβλήματος. Για παράδειγμα, σε ένα πρόγραμμα κοινωνικής ασφάλισης οι ηλικίες των ασφαλισμένων θα κυμαίνονται από **18** έως **65**, διότι υποθέτουμε πως οι εργαζόμενοι είναι ενήλικες που εργάζονται μέχρι τα **65** τους χρόνια. Αντίθετα στο πρόγραμμα καταχώρησης στοιχείων ασθενών σε ένα νοσοκομείο, η ηλικία του ασθενή μπορεί να είναι οποιαδήποτε.

Σε ορισμένες περιπτώσεις είναι δυνατό να καθορίσουμε όχι μόνο το εύρος των τιμών για ένα πεδίο, αλλά και **τη μορφή που θα έχουν οι τιμές που καταχωρούνται σε αυτό**. Για παράδειγμα, ένας αριθμός αυτοκινήτου αποτελείται **από τρία γράμματα και τέσσερα ψηφία** – π.χ. **TKE-6960**. Στην περίπτωση αυτή μπορούμε να γράψουμε τη μορφή των τιμών του πεδίου ως **TTT-DDDD** όπου το σύμβολο **T** υποδηλώνει **χαρακτήρα (text)** και παίρνει τιμές στο διάστημα **A-Z**, ενώ το σύμβολο **D**, υποδηλώνει **ψηφίο (digit)** και παίρνει τιμές στο διάστημα **0-9**. Άλλες φορές πάλι είναι δυνατό να καθορίσουμε και κάποιες επιπλέον πληροφορίες. Για παράδειγμα σε μια εφαρμογή που καταχωρούμε αριθμητικές τιμές για διάφορες περιπτώσεις πεδίων, ίσως χρειαστεί να καθορίσουμε και τη **μονάδα μέτρησης της τιμής**, κάτι που συνηθίζεται σε όλα σχεδόν τα προγράμματα που χρησιμοποιούνται στο χώρο των επιχειρήσεων για εφαρμογές μηχανογράφησης.

Ορισμός 1 : Ως **σχήμα μιας σχέσης (relation schema) R**, ορίζεται ένα **σύνολο πεδίων** $\{A_1, A_2, \dots, A_n\}$ κάθε ένα εκ των οποίων παίρνει τιμές μέσα από ένα **σύνολο τιμών (domain) D** – για κάθε πεδίο A_i , το πεδίο τιμών του, **D**, θα συμβολίζεται με **dom(A_i)**. Το πλήθος των πεδίων του συνόλου **R** ονομάζεται **βαθμός (degree)** της σχέσης, ενώ η συμβολοσειρά **R**, είναι το **όνομα** της σχέσης. Ένα παράδειγμα μιας σχέσης είναι ο πίνακας **STUDENT** που περιγράψαμε στην προηγούμενη παράγραφο, και ο οποίος σύμφωνα με τον παραπάνω ορισμό, μπορεί να γραφεί με τη μορφή

STUDENT {Name, SSN, HomePhone, Address, OfficePhone, Age, GPA}

Στη σύνταξη αυτή, **R = «STUDENT»** είναι το όνομα της σχέσης η οποία στην προκειμένη περίπτωση, έχει **επτά πεδία**, κάθε ένα εκ των οποίων, παίρνει τιμές μέσα από κάποιο σύνολο, **D**.

Ορισμός 2 : Ως **σχέση ή στιγμιότυπο σχέσης (relation instance) r**, του **σχήματος σχέσης R** ($A_1, A_2, \dots, A_n$) ορίζεται ένα σύνολο από πλειάδες $r = \{t_1, t_2, t_3,$

....., t_m }. Κάθε πλειάδα είναι μια **διατεταγμένη λίστα (ordered list)** από n τιμές $\langle v_1, v_2, \dots, v_n \rangle$, με την κάθε τιμή v_i ($1 \leq i \leq n$), να παίρνει τιμές από το πεδίο τιμών $\text{dom}(A_i)$, ενώ εναλλακτικά μπορεί να πάρει και την τιμή **NULL**. Ένα στιγμιότυπο r ενός σχήματος σχέσης R , μπορεί να γραφεί και ως $r(R)$.

Επειδή το κάθε στιγμιότυπο σχέσης αποτελείται από ένα σύνολο n πεδίων A_i κάθε ένα εκ των οποίων παίρνει τιμές μέσα από ένα σύνολο τιμών $\text{dom}(A_i)$, μπορούμε να θεωρήσουμε πως **το σύνολο των πλειάδων που συνιστά το στιγμιότυπο σχέσης $r(R)$, είναι υποσύνολο του καρτεσιανού γινομένου των συνόλων $\text{dom}(A_i)$, και επομένως θα ισχύει η συνθήκη**

$$r(R) \subseteq (\text{dom}(A_1) \times \text{dom}(A_2) \times \dots \times \text{dom}(A_n))$$

Είναι σημαντικό να αναφερθεί στο σημείο αυτό πως η **διάταξη (ordering)** των πλειάδων σε μια σχέση, δεν έχει καμιά σημασία. Αυτό γίνεται διότι μια σχέση ορίζεται ως **ένα σύνολο από πλειάδες**, και σύμφωνα με τον ορισμό του συνόλου, τα στοιχεία του δε χρειάζεται να είναι **διατεταγμένα**. Αντίθετα, μια πλειάδα, αποτελείται από ένα πλήθος **διατεταγμένων πεδίων**, και επομένως η σειρά με την οποία αναγράφονται οι τιμές μιας πλειάδας είναι σημαντική. Χρησιμοποιώντας εναλλακτικούς ορισμούς για το στιγμιότυπο σχέσης, μπορούμε να επανακαθορίσουμε τα βασικά της χαρακτηριστικά έτσι ώστε να άρουμε την προϋπόθεση της διάταξης των πεδίων μιας πλειάδας – αυτό το θέμα όμως δεν θα μας απασχολήσει περισσότερο.

Ολοκληρώνουμε αυτή τη συνοπτική περιγραφή των βασικών χαρακτηριστικών του σχεσιακού μοντέλου, με την παρατήρηση, πως **οι τιμές που καταχωρούνται σε μια πλειάδα, θα πρέπει να είναι ατομικές (atomic)** να μην υποδιαιρούνται δηλαδή σε μικρότερες μονάδες πληροφορίας. Σύνθετες τιμές δεδομένων, δεν μπορούν να χρησιμοποιηθούν αλλά θα πρέπει να γραφούν με τρόπο που να μην παραβιάζει την παραπάνω αρχή. Αυτός ο τρόπος φορμαλισμού ενός πίνακα έτσι ώστε όλα του τα πεδία να έχουν ατομικές τιμές, αναφέρεται και ως **πρώτη κανονική μορφή (1st Normal Form, 1NF)** και θα μελετηθεί στο επόμενο κεφάλαιο. Τέλος, όπως έχουμε ήδη αναφέρει, **για όσα πεδία δεν γνωρίζουμε ή δεν διαθέτουμε την τιμή που τα χαρακτηρίζει, χρησιμοποιούμε για αυτά την τιμή NULL**.

ΠΕΔΙΑ – ΚΛΕΙΔΙΑ : Ένα από τα βασικά χαρακτηριστικά μιας σχέσης είναι η μοναδικότητα κάθε μιας από τις πλειάδες που περιέχει. Αυτό σημαίνει πως **όλες οι πλειάδες είναι διαφορετικές μεταξύ τους** και σε καμιά περίπτωση δεν πρόκειται να καταχωρήσουμε δυο φορές την ίδια πλειάδα. Επειδή όμως μια πλειάδα είναι ένα σύνολο πεδίων, μπορούμε γενικά να θεωρήσουμε πως σε κάθε μια από τις πλειάδες ενός πίνακα, υπάρχει ένας συνδυασμός πεδίων με την ακόλουθη ιδιότητα: **ο συνδυασμός των τιμών αυτών των πεδίων είναι μοναδικός για κάθε εγγραφή**. Το σύνολο αυτών των πεδίων ονομάζεται **superkey** του σχεσιακού σχήματος στο οποίο αναφερόμαστε.

Από τον παραπάνω ορισμό είναι προφανές πως **η κάθε πλειάδα διαθέτει τουλάχιστον ένα superkey : το σύνολο των πεδίων που περιέχει**. Εάν θεωρήσουμε τώρα ένα **superkey** τέτοιο ώστε η αφαίρεση έστω και ενός από τα πεδία του να άρει την ιδιότητα της μοναδικότητας της τιμής του – δηλαδή η τιμή του να μην είναι πλέον

μοναδική για κάθε πλειάδα – τότε αυτό το **superkey** ονομάζεται **πεδίο κλειδί (key)** του σχεσιακού σχήματος. Με άλλα λόγια, **το πεδίο κλειδί μιας σχέσης είναι ένα superkey από το οποίο δεν μπορούμε να αφαιρέσουμε κανένα πεδίο (minimal superkey).**

Από τα όσα αναφέραμε στις προηγούμενες παραγράφους, γίνεται εύκολα αντιληπτό πως **η τιμή του πεδίου κλειδιού είναι μοναδική για κάθε πλειάδα**, και επομένως μπορεί να χρησιμοποιηθεί για να ξεχωρίσουμε τις πλειάδες μιας σχέσης μεταξύ τους. Ας σημειωθεί πως **στη γενική περίπτωση μπορεί να υπάρχουν περισσότερα από ένα πεδία που να μπορούν να χρησιμοποιηθούν ως πεδία κλειδιά μιας σχέσης**. Τα πεδία αυτά ονομάζονται **υποψήφια κλειδιά (candidate keys)** και ένα από αυτά επιλέγεται για να γίνει το **πρωτεύον κλειδί (primary key)** του σχήματος της σχέσης. Όσον αφορά το πιο από τα πεδία της σχέσης θα επιλεγεί για να γίνει το πρωτεύον κλειδί, αυτό εξαρτάται από τη μορφή της πληροφορίας που διατηρούμε για κάθε σχέση. Στη γενική περίπτωση ως **πρωτεύον κλειδί επιλέγουμε ένα πεδίο που να λαμβάνει ξεχωριστή τιμή για κάθε πλειάδα**, π.χ. αριθμός ταυτότητας, αριθμός αυτοκινήτου, κ.λ.π. Στις σελίδες που ακολουθούν και κάθε φορά που αναφερόμαστε στη δομή κάποιου πίνακα, το πρωτεύον κλειδί του πίνακα, θα είναι **υπογραμμισμένο**.

ΣΧΕΣΙΑΚΟ ΣΧΗΜΑ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ

Με βάση το **σχήμα** μιας σχέσης **R** που ορίσαμε στις προηγούμενες παραγράφους, μπορούμε τώρα να επεκτείνουμε τον παραπάνω ορισμό και να ορίσουμε το **σχεσιακό σχήμα μιας βάσης δεδομένων**. Έτσι ως **σχεσιακό σχήμα μιας βάσης δεδομένων (relational database schema)**, **S**, ορίζουμε ένα σύνολο σχημάτων σχέσεων **S = {R₁, R₂, ..., R_m}** και ένα σύνολο κανόνων ακεραιότητας (**integrity constraint**) για αυτά τα σχήματα. Ένα **στιγμιότυπο** του σχεσιακού σχήματος της βάσης δεδομένων (**relational database instance**) ορίζεται ως ένα σύνολο **στιγμιότυπων σχέσεων {r₁, r₂, ..., r_m}** με το κάθε στιγμιότυπο **r_i** της σχέσης **R_i** να υπακούει στους κανόνες ακεραιότητας του σχεσιακού σχήματος της βάσης. Το επόμενο σχήμα αναπαριστά το σχεσιακό σχήμα της βάσης δεδομένων της εταιρείας που μελετήσαμε στο προηγούμενο κεφάλαιο ενώ σε επόμενη σελίδα παρατίθεται και ένα ενδεικτικό **στιγμιότυπο** αυτού του σχήματος, που για κάθε έναν από τους πίνακες της βάσης περιλαμβάνει ένα πλήθος εγγραφών με συγκεκριμένες τιμές πεδίων.

EMPLOYEE

FNAME	MINIT	LNAME	<u>SSN</u>	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO
-------	-------	-------	------------	-------	---------	-----	--------	----------	-----

DEPARTMENT

DNAME	<u>DNUMBER</u>	MGRSSN	MGRSTARTDATE
-------	----------------	--------	--------------

DEPT_LOCATIONS

<u>DNUMBER</u>	<u>DLOCATION</u>
----------------	------------------

DEPENDENT

<u>ESSN</u>	<u>DEPENDENT-NAME</u>	SEX	BDATE	RELATIONSHIP
-------------	-----------------------	-----	-------	--------------

PROJECT

PNAME	<u>PNUMBER</u>	PLOCATION	DNUM
-------	----------------	-----------	------

WORKS_ON

<u>ESSN</u>	<u>PNO</u>	HOURS
-------------	------------	-------

Σχήμα 8 : Το σχεσιακό σχήμα για τη βάση δεδομένων της εταιρείας

KANONEΣ ΑΚΕΡΑΙΟΤΗΤΑΣ: Οι κανόνες ακεραιότητας (**integrity constraints**) ενός σχεσιακού σχήματος μιας βάσης δεδομένων, διασφαλίζουν τη συνέπεια των δεδομένων της βάσης, και ισχύουν για κάθε στιγμιότυπό της. Ο πρώτος από αυτούς τους κανόνες (**key constraint**) επιβάλλει την απόδοση **μοναδικών τιμών** σε όλα τα πεδία του πίνακα που έχουν χαρακτηριστεί ως **υποψήφια κλειδιά** (**candidate keys**). Με άλλα λόγια **δεν θα πρέπει να υπάρχουν στον ίδιο πίνακα, δύο εγγραφές που να έχουν την ίδια τιμή στα υποψήφια κλειδιά τους** – και προφανώς στο πρωτεύον κλειδί τους που είναι κάποιο από αυτά. Ο δεύτερος κανόνας (**entity integrity constraint**) **απαγορεύει την απόδοση της τιμής NULL, στο πρωτεύον κλειδί της κάθε εγγραφής του πίνακα**. Αυτό γίνεται επειδή ως γνωστόν **το πρωτεύον κλειδί μας επιτρέπει να ξεχωρίσουμε τις εγγραφές του πίνακα**, και επομένως εάν δύο εγγραφές έχουν τιμή **NULL** στο πρωτεύον κλειδί τους, δεν είναι δυνατή η διάκριση ανάμεσά τους. Τέλος, ο τρίτος κανόνας (**referential integrity constraint**) αναφέρεται στις **συσχετίσεις** που υφίστανται ανάμεσα στους πίνακες της βάσης. Ο κανόνας αυτός λέει ότι **εάν η τιμή κάποιου πεδίου μιας από τις εγγραφές ενός πίνακα αναφέρεται σε εγγραφή κάποιου άλλου πίνακα, αυτή θα πρέπει να είναι μια υπάρχουσα εγγραφή**. Ας πάρουμε για παράδειγμα την πρώτη εγγραφή του πίνακα **EMPLOYEE**. Αυτή η εγγραφή, στο πεδίο **DNO** έχει την τιμή **5**, κάτι που σημαίνει πως ο συγκεκριμένος υπάλληλος, εργάζεται στο τμήμα της εταιρείας με κωδικό αριθμό **5**. Στην περίπτωση αυτή ο τρίτος κανόνας ακεραιότητας **επιβάλλει την ύπαρξη στον πίνακα DEPARTMENT μιας εγγραφής το πρωτεύον κλειδί της οποίας να έχει την τιμή 5**. Εάν δεν υπάρχει αυτή η εγγραφή, τότε η εγγραφή στον πίνακα **EMPLOYEE** αναφέρεται σε κάποιο ανύπαρκτο τμήμα, με αποτέλεσμα τελικά να οδηγούμαστε σε μια βάση με **ασυνεπή δεδομένα (inconsistent data)**.

Η συσχέτιση ανάμεσα σε δύο πίνακες μιας σχεσιακής βάσης δεδομένων, μας επιτρέπει να ορίσουμε ένα επιπλέον τύπο κλειδιού, που ονομάζεται **ξένο κλειδί (foreign key)**. Ένα πεδίο ενός πίνακα **A** ονομάζεται **ξένο κλειδί**, όταν αποτελεί **πρωτεύον κλειδί** κάποιου άλλου πίνακα **B** και έχει τοποθετηθεί στον πίνακα **A** ως αποτέλεσμα της συσχέτισης που υφίσταται ανάμεσα στους πίνακες **A** και **B** σύμφωνα με το **μοντέλο οντοτήτων – συσχετίσεων**. Ας πάρουμε για παράδειγμα τη συσχέτιση ανάμεσα στους τύπους οντότητας **DEPARTMENT** και **EMPLOYEE**. Επειδή η συσχέτιση αυτή έχει **πολλαπλότητα 1:N** μπορούμε να την υλοποιήσουμε, τοποθετώντας το πρωτεύον κλειδί του πίνακα **DEPARTMENT** – που είναι το πεδίο

πρωτεύον κλειδί του πίνακα **DEPARTMENT** – που είναι το πεδίο **DNUMBER** – στον πίνακα **EMPLOYEE** (με το όνομα **DNO**). Στην περίπτωση αυτή

EMPLOYEE

ENAME	MINIT	LNNAME	SSN	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO
John	B	Smith	123456789	09-Jan-55	731 Fordhen, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	08-Dec-45	638 Vass, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	19-Jul-58	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	20-Jun-31	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	15-Sep-52	975 Five Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	31-Jul-62	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	29-Mar-59	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	10-Nov-27	450 Stone, Houston, TX	M	55000	NULL	1

DEPARTMENT

DNAME	DNUMBER	MGRSSN	MGRSTARTDATE
Research	5	333445555	22-May-78
Administration	4	987654321	01-Jan-85
Headquarters	1	888665555	19-Jun-71

WORKS_ON

ESSN	PNO	HOURS
123456789	1	32.5
123456789	2	7.5
666884444	3	40.0
453453453	1	20.0
453453453	2	20.0
333445555	2	10.0
333445555	3	10.0
333445555	10	10.0
333445555	20	10.0
999887777	30	30.0
999887777	10	10.0
987987987	10	35.0
987987987	30	5.0
987654321	30	20.0
987654321	10	15.0
888665555	20	NULL

DEPT_LOCATIONS

DNUMBER	DLOCATION
1	Houston
2	Stafford
5	Bellaire
5	Sugarland
5	Houston

PROJECT

PNAME	PNUMBER	PLOCATION	DNUM
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
Newbenefits	30	Stafford	4

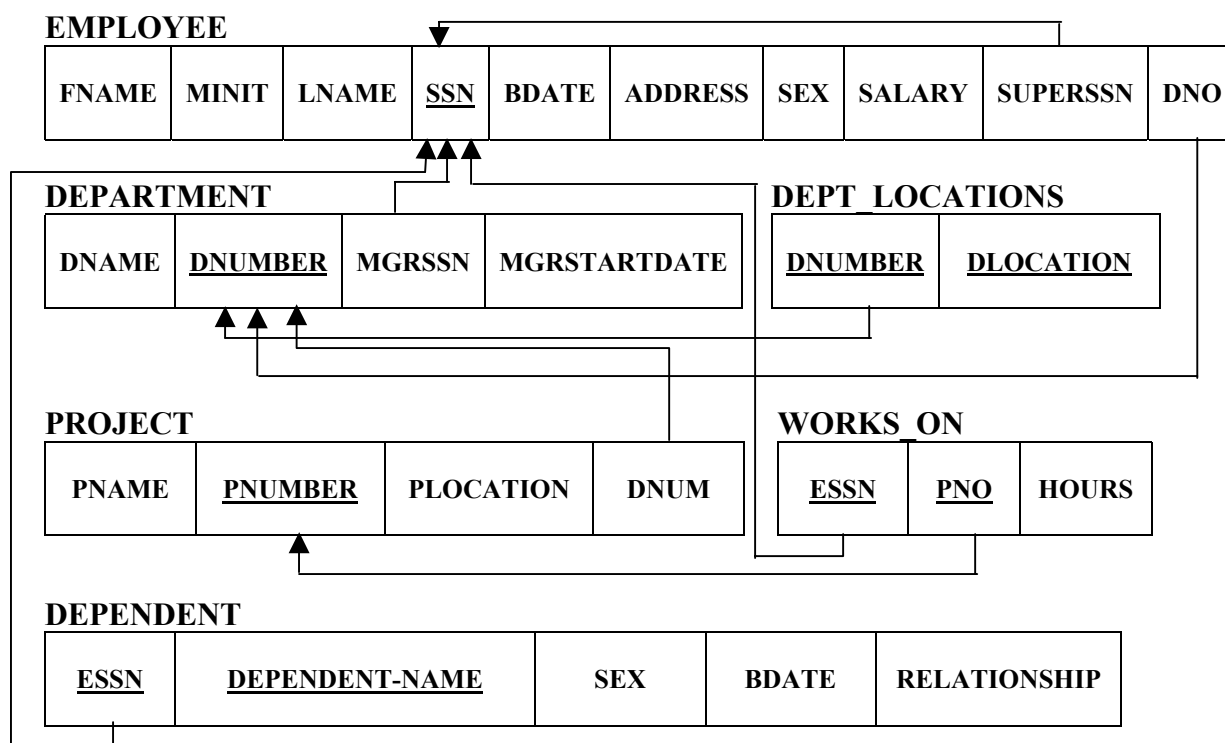
DEPENDENT

ESSN	DEPENDENT_NAME	SEX	BDATE	RELATIONSHIP
333445555	Alice	F	05-Apr-76	Daughter
333445555	Theodore	M	25-Oct-73	Son
333445555	Joy	F	03-May-48	Spouse
987654321	Ahner	M	29-Feb-32	Spouse
123456789	Michael	M	01-Jan-78	Son
123456789	Alice	F	31-Dec-78	Daughter
123456789	Elizabeth	F	05-May-57	Spouse

ο τρίτος κανόνας ακεραιότητας μπορεί να διατυπωθεί με τη μορφή **EMPLOYEE.DNO = DEPARTMENT.DNUMBER** και επομένως, η τιμή του πεδίου **DNO** σε κάθε εγγραφή του πίνακα **EMPLOYEE** θα πρέπει να αναφέρεται σε κάποια υπάρχουσα εγγραφή του πίνακα **DEPARTMENT**.

Εάν η συσχέτιση ανάμεσα σε τύπους οντότητας είναι **αναδρομική (recursive)**, ορίζεται δηλαδή ανάμεσα σε δύο αντίγραφα του ίδιου τύπου οντότητας, τότε, **το ξένο κλειδί του πίνακα, αναφέρεται σε κάποια εγγραφή του ίδιου πίνακα**. Αυτό συμβαίνει για παράδειγμα με το πεδίο **SUPERSSN** του πίνακα **EMPLOYEE** που αναφέρεται στον **επόπτη (supervisor)** του κάθε υπαλλήλου της εταιρείας. Αυτός ο επόπτης όμως, είναι και ο ίδιος υπάλληλος της εταιρείας, και επομένως η εγγραφή που αναφέρεται σε αυτόν, βρίσκεται και αυτή στον πίνακα **EMPLOYEE**. Αυτό σημαίνει πως **το πεδίο SUPERSSN είναι ξένο κλειδί που αναφέρεται σε κάποια εγγραφή η οποία βρίσκεται στον ίδιο πίνακα** και η οποία ασφαλώς θα πρέπει να υπάρχει σύμφωνα με τον τρίτο κανόνα ακεραιότητας. Για παράδειγμα το πεδίο **SUPERSSN** του υπαλλήλου **John Smith** έχει τιμή **333445555**. Αυτό σημαίνει πως θα πρέπει στον ίδιο πίνακα να υπάρχει κάποιος εργαζόμενος με **SSN** ίσο με **333445555** – αυτός ο εργαζόμενος πράγματι υπάρχει και είναι ο **Franklin Wong**, και επομένως στην περίπτωση αυτή, ο τρίτος κανόνας ακεραιότητας ικανοποιείται.

Ένας απλός και εύκολος τρόπος για να ορίσουμε τις συσχετίσεις ανάμεσα στα πρωτεύοντα και στα ξένα κλειδιά των πινάκων σύμφωνα με τον τρίτο κανόνα ακεραιότητας, είναι να τροποποιήσουμε την εικόνα του σχεσιακού σχήματος της βάσης προσθέτοντας προσανατολισμένες ευθείες που να ξεκινούν από το ξένο κλειδί κάποιου πίνακα και να καταλήγουν στο αντίστοιχο πρωτεύον κλειδί του πίνακα με το οποίο συσχετίζονται. Αυτή η εικόνα για το σχεσιακό σχήμα της βάσης δεδομένων της εταιρείας παρουσιάζεται στην επόμενη εικόνα. (Σχήμα 9).



ΠΑΡΑΔΕΙΓΜΑΤΑ ΕΦΑΡΜΟΓΗΣ ΤΩΝ ΚΑΝΟΝΩΝ ΑΚΕΡΑΙΟΤΗΤΑΣ

Από την παραπάνω ανάλυση είναι προφανές πως **κάθε φορά που λαμβάνει χώρα απόπειρα τροποποίησης του περιεχομένου της βάσης δεδομένων, θα πρέπει να διασφαλίζεται η τήρηση των τριών κανόνων ακεραιότητας που έχουμε ορίσει στην προηγούμενη ενότητα**. Σε όλες περιπτώσεις, η τροποποίηση του περιεχομένου μιας βάσης θα περιλαμβάνει διαδικασίες **εισαγωγής νέων εγγραφών (insert operation)**, **διαγραφής υπαρχόντων εγγραφών των πινάκων (delete operation)** ή **τροποποίησης του περιεχομένου αυτών των εγγραφών (update operation)**. Κάθε φορά που ο χρήστης προσπαθεί να πραγματοποιήσει αυτές τις τρεις κατηγορίες ενεργειών, το σύστημα διαχείρισης βάσεων δεδομένων, ελέγχει εάν αυτές οι ενέργειες παραβιάζουν ή όχι τους κανόνες ακεραιότητας του σχήματος της βάσης, και επιτρέπει την εκτέλεσή τους, μόνο όταν αυτοί οι κανόνες δεν παραβιάζονται. Στις επόμενες σελίδες παρουσιάζονται παραδείγματα εισαγωγής, διαγραφής και τροποποίησης εγγραφών των πινάκων για τη βάση δεδομένων της εταιρείας, και εφαρμογή επί αυτών, των κανόνων ακεραιότητας, προκειμένου να ελεγχθεί εάν αυτοί παραβιάζονται ή τηρούνται από αυτές τις ενέργειες.

Παραδείγματα εισαγωγής εγγραφών (insert operation)

Insert <'Cecilia', 'F', 'Kolonsky', '677678989', '05-Apr-50', '6357 Windy Lane, Katy, TX', 'F', 28000, NULL, 4> into EMPLOYEE

Αυτή η διαδικασία δεν παραβιάζει κανέναν από τους κανόνες ακεραιότητας και είναι αποδεκτή από το σύστημα.

Insert <'Alicia', 'J', 'Zelaya', '999887777', '05-Apr-50', '6357 Windy Lane, Katy, TX', 'F', 28000, '987654321', 4> into EMPLOYEE

Αυτή η διαδικασία παραβιάζει **τον πρώτο κανόνα ακεραιότητας (key constraint)** επειδή υπάρχει ήδη στον πίνακα EMPLOYEE κάποιος άλλος υπάλληλος με το ίδιο **πρωτεύον κλειδί (SSN)**. Για το λόγο αυτό δεν γίνεται αποδεκτή από το σύστημα.

Insert <'Cecilia', 'F', 'Kolonsky', NULL, '05-Apr-50', '6357 Windy Lane, Katy, TX', 'F', 28000, NULL, 4> into EMPLOYEE

Αυτή η διαδικασία παραβιάζει **το δεύτερο κανόνα ακεραιότητας (entity constraint)** διότι η εγγραφή που επιχειρούμε να καταχωρήσουμε έχει τιμή **NULL** στο πρωτεύον κλειδί της. Για το λόγο αυτό δεν γίνεται αποδεκτή από το σύστημα.

Insert <'Cecilia', 'F', 'Kolonsky', '677678989', '05-Apr-50', '6357 Windy Lane, Katy, TX', 'F', 28000, '987654321', 7> into EMPLOYEE

Αυτή η διαδικασία παραβιάζει **τον τρίτο κανόνα ακεραιότητας** διότι αναφέρεται σε τμήμα της εταιρείας με κωδικό αριθμό 7, αλλά στον πίνακα **DEPARTMENT** δεν υπάρχει κάποια εγγραφή το πρωτεύον κλειδί της οποίας να έχει την τιμή 7. Για το λόγο αυτό δεν γίνεται αποδεκτή από το σύστημα.

Παραδείγματα διαγραφής εγγραφών (delete operation)

DELETE the WORKS_ON tuple with SSN='999887777' AND PNO = 10

Αυτή η διαδικασία δεν παραβιάζει κανέναν από τους κανόνες ακεραιότητας και είναι αποδεκτή από το σύστημα.

DELETE the EMPLOYEE tuple with SSN='999887777'

Αυτή η διαδικασία δεν είναι αποδεκτή, διότι υπάρχουν εγγραφές στον πίνακα **WORKS_ON** που αναφέρονται στην εγγραφή που επιθυμούμε να διαγράψουμε, και επομένως η διαγραφή της θα οδηγήσει σε παραβίαση του τρίτου κανόνα ακεραιότητας (referential integrity constraint).

DELETE the EMPLOYEE tuple with SSN = '333445555'

Αυτή η διαδικασία δεν είναι αποδεκτή, διότι υπάρχουν εγγραφές στους πίνακες **EMPLOYEE**, **DEPARTMENT**, **WORKS_ON** και **DEPENDENT** που αναφέρονται στην εγγραφή που επιθυμούμε να διαγράψουμε, και επομένως η διαγραφή της θα οδηγήσει σε παραβίαση του τρίτου κανόνα ακεραιότητας (referential integrity constraint).

Από τους τρεις κανόνες ακεραιότητας που περιγράψαμε στις προηγούμενες σελίδες, η διαδικασία της διαγραφής εγγραφών από κάποιο πίνακα, παραβιάζει μόνο τον τρίτο κανόνα, σε περιπτώσεις κατά τις οποίες επιχειρούμε να διαγράψουμε μια εγγραφή στην οποία αναφέρονται εγγραφές άλλων πινάκων τις βάσης. Στις περιπτώσεις αυτές έχουμε γενικά τρεις επιλογές: (α) να απαγορεύσουμε τη διαγραφή αυτής της πλειάδας, (β) να διαγράψουμε εκτός από την εν λόγω πλειάδα και όλες τις πλειάδες των άλλων πινάκων που αναφέρονται σε αυτή (cascade delete operation) και (γ) να μη διαγράψουμε τις πλειάδες των άλλων πινάκων αλλά να τροποποιήσουμε το περιεχόμενό τους τοποθετώντας τιμές NULL στα πεδία που αναφέρονται στην πλειάδα που επιχειρούμε να διαγράψουμε (referencing attributes). Ας σημειωθεί πως για να είναι δυνατή η πραγματοποίηση της τρίτης επιλογής δεν θα πρέπει να πεδία που θα λάβουν την τιμή NULL να αποτελούν τμήμα του πρωτεύοντος κλειδιού του πίνακα, διότι στην περίπτωση αυτή θα παραβιαστεί ο πρώτος κανόνας ακεραιότητας (key constraint).

Παραδείγματα τροποποίησης εγγραφών (modify operation)

**MODIFY the SALARY of the EMPLOYEE tuple
with SSN = '999887777' to 28000**

Αυτή η διαδικασία δεν παραβιάζει κανέναν από τους κανόνες ακεραιότητας και είναι αποδεκτή από το σύστημα.

MODIFY the DNO of the EMPLOYEE tuple with SSN = '999887777' to 1

Αυτή η διαδικασία δεν παραβιάζει κανέναν από τους κανόνες ακεραιότητας και είναι αποδεκτή από το σύστημα.

MODIFY the DNO of the EMPLOYEE tuple with SSN = '999887777' to 7

Αυτή η διαδικασία παραβιάζει τον τρίτο κανόνα ακεραιότητας (referential integrity constraint) και δεν γίνεται αποδεκτή από το σύστημα

**MODIFY the SSN of the EMPLOYEE tuple with
SSN = '999887777' to '987654321'**

Αυτή η διαδικασία παραβιάζει τον πρώτο κανόνα ακεραιότητας (key constraint) και δεν γίνεται αποδεκτή από το σύστημα.

ΑΠΕΙΚΟΝΙΣΗ ΤΟΥ ΜΟΝΤΕΛΟΥ ΟΝΤΟΤΗΤΩΝ ΣΥΣΧΕΤΙΣΕΩΝ ΣΤΟ ΣΧΕΣΙΑΚΟ ΜΟΝΤΕΛΟ

Έχοντας περιγράψει το μοντέλο οντοτήτων συσχετίσεων και το σχεσιακό μοντέλο βάσεων δεδομένων, μπορούμε τώρα να μελετήσουμε τον τρόπο με τον οποίο είναι δυνατή η δημιουργία του σχεσιακού μοντέλου μιας βάσης δεδομένων, από το μοντέλο οντοτήτων συσχετίσεων που έχουμε δημιουργήσει για αυτή τη βάση. Αυτή η διαδικασία είναι γνωστή ως **απεικόνιση (mapping)** του ενός μοντέλου στο άλλο και είναι ένα από τα πιο σημαντικά στάδια του λογικού σχεδιασμού μιας βάσης δεδομένων.

Η παρουσίαση του αλγόριθμου μετατροπής του ενός μοντέλου στο άλλο, θα γίνει και πάλι δια της χρήσης του παραδείγματος της **βάσης δεδομένων της εταιρείας (company database)**. Το σχήμα αυτής της βάσης, όπως έχουμε δει σε προηγούμενο κεφάλαιο, περιλαμβάνει **τέσσερις τύπους οντότητας (EMPLOYEE, DEPARTMENT, PROJECT, DEPENDENT** (με τον τελευταίο τύπο οντότητας να είναι **weak entity**)) και **πέντε τύπους συσχέτισης (WORKS_FOR, SUPERVISION, DEPENDENTS_OF** (με **πολλαπλότητα 1:N**), **MANAGES** (με **πολλαπλότητα 1:1**), **WORKS_ON** (με **πολλαπλότητα M:N**)). Για κάθε τύπο οντότητας ορίσαμε το **προτεύον κλειδί** ενώ σε κάθε περίπτωση εκτός από τα **απλά πεδία**, εντοπίσαμε τόσο **σύνθετα πεδία (composite attributes)** όσο και **πεδία πολλαπλών τιμών (multivalued attributes)**. Όσον αφορά τον τρόπο αλληλεπίδρασης ανάμεσα στους τύπους οντότητας της βάσης, αυτός σε γενικές γραμμές, έχει ως εξής :

Κάθε υπάλληλος εργάζεται μόνο σε ένα τμήμα αλλά μπορεί να απασχολείται σε πολλά έργα (projects) ταυτόχρονα. Μπορεί να έχει υπό την εποπτεία του πολλούς υπαλλήλους αλλά ο ίδιος παρακολουθείται από ένα μόνο επόπτη. Τέλος είναι δυνατό να **εποπτεύει** ο ίδιος κάποιο τμήμα (αλλά **μόνο ένα τμήμα κάθε φορά**) και μπορεί να έχει **πολλά προστατευόμενα μέλη**. Τα στοιχεία που κρατούμε για κάθε υπάλληλο, είναι **το όνομά του, ο κωδικός SSN, η ημερομηνία γέννησής του, το φύλο του, και ο μισθός του.**

Κάθε τμήμα περιγράφεται **από ένα όνομα, ένα κωδικό και ένα σύνολο από τοποθεσίες στις οποίες είναι εγκατεστημένο.** Ένα τμήμα διευθύνεται από ένα και μοναδικό **manager** αλλά μπορεί να διαθέτει **πολλούς υπαλλήλους** που να εργάζονται για αυτό. Τέλος το κάθε ένα από τα τμήματα της εταιρείας, μπορεί γενικά να αναλάβει **πολλά έργα ταυτόχρονα.**

Κάθε έργο παρακολουθείται από ένα και μόνο τμήμα αλλά πραγματοποιείται από πολλούς υπαλλήλους **ταυτόχρονα**. Ένα έργο περιγράφεται **από το όνομά του, τον κωδικό του, και την τοποθεσία στην οποία πραγματοποιείται**.

Τέλος **το κάθε προστατευόμενο μέλος συσχετίζεται με ένα και μοναδικό υπάλληλο**.

Με βάση την παραπάνω περιγραφή όσον αφορά τη δομή και τη λειτουργία της εταιρείας, κατασκευάσαμε **το μοντέλο οντοτήτων συσχετίσεων** για αυτό το σύστημα (σελ. 27) ενώ σε προηγούμενη ενότητα δημιουργήσαμε και το αντίστοιχο **σχεσιακό σχήμα της βάσης δεδομένων** (σελ. 35). Η απεικόνιση του ενός μοντέλου στο άλλο πραγματοποιείται δια της εφαρμογής μιας διαδικασίας **έξι βημάτων**, τα οποία σε γενικές γραμμές είναι τα ακόλουθα :

ΒΗΜΑ 1 : Για κάθε τύπο οντότητας **E** του μοντέλου οντοτήτων συσχετίσεων, δημιουργούμε ένα πίνακα **R** που να περιλαμβάνει **όλα τα απλά πεδία** του τύπου οντότητας. Τα **πεδία πολλαπλών τιμών (multivalued attributes)** δεν χρησιμοποιούνται ακόμη. Εάν κάποιο πεδίο του **E** είναι **σύνθετο** προσθέτουμε μόνο τα απλά πεδία από τα οποία αποτελείται. Στη συνέχεια επιλέγουμε κάποιο από τα πεδία κλειδιά του **E** και το ορίζουμε ως **το πρωτεύον κλειδί του πίνακα R**. Εάν αυτό το πεδίο είναι σύνθετο, τότε και το πρωτεύον κλειδί του πίνακα θα είναι σύνθετο κλειδί, και θα αποτελείται από περισσότερα από ένα πεδία. Στη δομή των πινάκων της βάσης που παρουσιάζεται στις επόμενες σελίδες, τα πρωτεύοντα κλειδιά των πινάκων εμφανίζονται υπογραμμισμένα.

Στο παράδειγμά μας, από τους τρεις τύπους οντότητας **EMPLOYEE**, **DEPARTMENT** και **PROJECT** (ο τύπος οντότητας **DEPENDENT** είναι **weak entity type** και θα αναλυθεί αργότερα) δημιουργούμε τους τρεις αντίστοιχους πίνακες (που φέρουν τα ίδια ονόματα). Με βάση τα όσα αναφέραμε στην προηγούμενη παράγραφο αυτοί οι πίνακες θα έχουν την ακόλουθη δομή.

EMPLOYEE {FNAME, MINIT, LNAME, SSN, BDATE, ADDRESS, SALARY}

DEPARTMENT {DNAME, DNUMBER}

PROJECT {PNAME, PNUMBER, PLOCATION}

ΒΗΜΑ 2 : Για κάθε αδύναμο τύπο οντότητας (**weak entity type**) **W** του μοντέλου οντοτήτων συσχετίσεων, εντοπίζουμε την **οντότητα κάτοχο (owner entity)** **E** από την οποία εξαρτάται. Στη συνέχεια για τον τύπο οντότητας **W**, δημιουργούμε ένα πίνακα **R**, ο οποίος περιλαμβάνει **όλα τα απλά πεδία του W**. Εάν ο τύπος **W** περιέχει **σύνθετα πεδία (composite attributes)**, τότε όπως και στην προηγούμενη περίπτωση προσθέτουμε μόνο τα απλά πεδία από τα οποία αυτά αποτελούνται. Στη συνέχεια προσθέτουμε στον πίνακα **R** το **πρωτεύον κλειδί** του πίνακα που έχουμε ήδη δημιουργήσει στο **Βήμα 1** για την οντότητα κάτοχο του **W**. Επειδή αυτό το πεδίο είναι **πρωτεύον κλειδί** σε κάποιο άλλο πίνακα, θα συμμετάσχει στον πίνακα **R**, ως **ξένο κλειδί (foreign key)**. Το πρωτεύον κλειδί του πίνακα **R**, θα είναι **σύνθετο**, και θα αποτελείται **από το συνδυασμό του ξένου κλειδιού που θα προσθέσουμε**

και του μερικού κλειδιού (partial key) που έχουμε ήδη ορίσει κατά το στάδιο δημιουργίας του μοντέλου οντοτήτων συσχετίσεων.

Στο παράδειγμά μας έχουμε ένα μόνο αδύναμο τύπο οντότητας, τον **DEPENDENT**, για τον οποίο η οντότητα κάτοχος είναι ο **EMPLOYEE**. Δημιουργούμε λοιπόν ένα νέο πίνακα με το όνομα **DEPENDENT** και προσθέτουμε σε αυτόν όλα τα απλά πεδία του αντίστοιχου τύπου οντότητας. Από το διάγραμμα οντοτήτων συσχετίσεων, διαπιστώνουμε πως τα πεδία αυτά είναι τα {**Name**, **Sex**, **BirthDate**}. Στη συνέχεια προσθέτουμε το πρωτεύον κλειδί του πίνακα **EMPLOYEE** που έχουμε δημιουργήσει στο **Βήμα 1** για την οντότητα κάτοχο του **DEPENDENT** που είναι ο **EMPLOYEE**. Το πεδίο αυτό είναι το **SSN** αλλά στον πίνακα **DEPENDENT** θα προστεθεί με το όνομα **ESSN**. Επειδή το partial key του **DEPENDENT** είναι το **DEPENDENT_NAME**, έπεται πως το πρωτεύον κλειδί του πίνακα **DEPENDENT** θα είναι το σύνθετο κλειδί **ESSN + DEPENDENT_NAME**. Έτσι η δομή του πίνακα **DEPENDENT** θα είναι η ακόλουθη :

DEPENDENT {ESSN, DEPENDENT_NAME, SEX, BDATE}

ΒΗΜΑ 3 : Για κάθε τύπο συσχέτισης **R**, η οποία έχει βαθμό 2 (ορίζεται δηλαδή ανάμεσα σε δύο οντότητες) και **πολλαπλότητα 1:1**, εντοπίζουμε τους πίνακες που έχουμε δημιουργήσει στο **Βήμα 1**, για τους δύο τύπους οντότητας που συμμετέχουν σε αυτή τη σχέση. Έστω ότι αυτοί οι πίνακες φέρουν τα ονόματα **R** και **S**. Στη συνέχεια, **διαλέγουμε τον ένα από τους δύο πίνακες (για παράδειγμα τον R) και προσθέτουμε σε αυτόν, το πρωτεύον κλειδί του πίνακα S**. Είναι προφανές πως αυτό το πεδίο θα συμμετέχει στον πίνακα **R** ως ξένο κλειδί. Επίσης στον πίνακα **R**, θα προσθέσουμε εκτός από το ξένο κλειδί και όλα τα πεδία που ενδεχομένως έχουμε ορίσει για τον τύπο συσχέτισης **R (relationship attributes)**.

Το ερώτημα που τίθεται στο σημείο αυτό αφορά το κριτήριο επιλογής του πίνακα στον οποίο θα προσθέσουμε τα επιπλέον πεδία. Στις πιο πολλές περιπτώσεις αυτό το κριτήριο έχει να κάνει με τον τύπο συμμετοχής του κάθε τύπου οντότητας στον τύπο συσχέτισης **R**. Έτσι είναι κοινή πρακτική να επιλέγουμε εκείνο τον πίνακα, για τον οποίο ο αντίστοιχος τύπος οντότητας χαρακτηρίζεται από **ολική συμμετοχή (total participation)** στον θεωρούμενο τύπο συσχέτισης. Εάν και οι δύο τύποι οντότητας παρουσιάζουν ολική συμμετοχή και επιπλέον δεν συμμετέχουν σε άλλο τύπο συσχέτισης, μπορούμε εναλλακτικά **να ενοποιήσουμε τους αντίστοιχους πίνακες και τα πεδία του τύπου συσχέτισης σε ένα και μοναδικό πίνακα**.

Στο παράδειγμά μας έχουμε μόνο ένα τύπο συσχέτισης με πολλαπλότητα **1:1** που φέρει το όνομα **MANAGES**, ορίζεται ανάμεσα στους τύπους οντότητας **EMPLOYEE** και **DEPARTMENT**, και επιπλέον περιέχει και ένα πεδίο, με το όνομα **StartDate**. Επειδή ο **EMPLOYEE** χαρακτηρίζεται από μερική συμμετοχή σε αυτόν τον τύπο συσχέτισης, ενώ αντίθετα το **DEPARTMENT** χαρακτηρίζεται από ολική συμμετοχή, προσθέτουμε στον πίνακα **DEPARTMENT**, τόσο το πρωτεύον κλειδί του πίνακα **EMPLOYEE** που είναι το **SSN** (αλλά με το όνομα **MGRSSN**) όσο και το πεδίο **StartDate** του τύπου συσχέτισης **MANAGES** (αλλά με το όνομα **MgrStartDate**). Έτσι τελικά ο πίνακας **DEPARTMENT** θα αποκτήσει τη δομή

DEPARTMENT {NAME, DNUMBER, MGRSSN, MGRSTARTDATE}

ΒΗΜΑ 4 : Για κάθε τύπο συσχέτισης **R** η οποία έχει **βαθμό 2** και **πολλαπλότητα 1:N**, εντοπίζουμε τον πίνακα που έχουμε δημιουργήσει (στο **Βήμα 1**) για τον τύπο οντότητας που βρίσκεται στην «**N πλευρά**» της συσχέτισης. Σε αυτόν τον πίνακα, προσθέτουμε ως **ξένο κλειδί**, το **πρωτεύον κλειδί** του πίνακα που αντιστοιχεί στον άλλο τύπο οντότητας (αυτόν που βρίσκεται στην «**1 πλευρά**» της συσχέτισης). Στο παράδειγμά μας, έχουμε τρεις τύπους συσχέτισης με πολλαπλότητα **1:N**. Τον τύπο συσχέτισης **WORKS_FOR** ανάμεσα στους τύπους οντότητας **DEPARTMENT** και **EMPLOYEE**, τον τύπο συσχέτισης **CONTROLS** ανάμεσα στους τύπους οντότητας **DEPARTMENT** και **PROJECT** και τον τύπο συσχέτισης **SUPERVISION** που είναι **αναδρομικός** και ορίζεται ανάμεσα σε δύο αντίγραφα του τύπου οντότητας **EMPLOYEE**. Η εφαρμογή του **Βήματος 4** σε αυτούς τους τύπους συσχέτισης, θα έχει τα ακόλουθα αποτελέσματα.

Στον τύπο συσχέτισης **WORKS_FOR**, ένα **DEPARTMENT** διαθέτει πολλούς **EMPLOYEES**. Σύμφωνα με τον παραπάνω αλγόριθμο, θα πρέπει να προσθέσουμε το πρωτεύον κλειδί του πίνακα **DEPARTMENT** στον πίνακα **EMPLOYEE** (στην προσθήκη αυτή το πεδίο **DNUMBER** προστίθεται με το όνομα **DNO**). Μετά την εφαρμογή αυτού του βήματος η δομή του πίνακα **EMPLOYEE** θα λάβει τη μορφή

**EMPLOYEE {FNAME, MINIT, LNAME, SSN, BDATE,
ADDRESS, SALARY, DNO}**

Στον τύπο συσχέτισης **SUPERVISION**, ένας **MANAGER** εποπτεύει πολλούς υπαλλήλους. Αυτό σημαίνει πως θα πρέπει να προσθέσουμε το πρωτεύον κλειδί του πίνακα που αντιπροσωπεύει τον **MANAGER** στον πίνακα **EMPLOYEE**. Επειδή όμως ο **MANAGER** είναι και αυτός **EMPLOYEE**, αυτό σημαίνει πως θα πρέπει να προσθέσουμε το πεδίο **SSN** – που είναι το πρωτεύον κλειδί του **EMPLOYEE** – στον πίνακα **EMPLOYEE**. Κατά την προσθήκη αυτή το πεδίο **SSN** προστίθεται στον πίνακα **EMPLOYEE** με το όνομα **SUPERSSN**. Μετά από αυτή την διαδικασία, ο πίνακας **EMPLOYEE** θα αποκτήσει τη δομή

**EMPLOYEE {FNAME, MINIT, LNAME, SSN, BDATE,
ADDRESS, SALARY, SUPERSSN, DNO}**

Τέλος, στον τύπο συσχέτισης **CONTROLS**, ένα **DEPARTMENT** ελέγχει γενικά πολλά **PROJECTS**. Αυτό σημαίνει πως θα πρέπει να προσθέσουμε το πρωτεύον κλειδί του πίνακα **DEPARTMENT** στον πίνακα **PROJECT**. Η προσθήκη αυτού του πεδίου γίνεται με το όνομα **DNO**. Έτσι, η δομή του πίνακα **PROJECT** θα λάβει τη μορφή

PROJECT {PNAME, PNUMBER, PLOCATION, DNO}

ΒΗΜΑ 5 : Για κάθε τύπο συσχέτισης με πολλαπλότητα **M:N**, δημιουργούμε ένα πίνακα **R**, στον οποίο προσθέτουμε ως **ξένα κλειδιά**, τα **πρωτεύοντα κλειδιά** των πινάκων που έχουμε δημιουργήσει (στο **Βήμα 1**) για τους τύπους οντότητας που συμμετέχουν σε αυτόν τον τύπο συσχέτισης. Επίσης προσθέτουμε και όλα τα πεδία που ενδεχομένως έχουμε ορίσει για αυτή τη συσχέτιση (**relationship attributes**). Το πρωτεύον κλειδί του πίνακα **R** που δημιουργήσαμε, είναι **σύνθετο κλειδί** και αποτε-

λείται από το συνδυασμό των πρωτεύοντων κλειδιών των δύο πινάκων που έχουμε προσθέσει στον πίνακα **R**.

Στο παράδειγμά μας, η μοναδική σχέση που υφίσταται με πολλαπλότητα **M:N**, είναι η σχέση **WORKS_ON**, που συσχετίζει τον τύπο οντότητας **EMPLOYEE** με τον τύπο οντότητας **PROJECT**. Τα πρωτεύοντα κλειδιά των ομώνυμων πινάκων που αναπαριστούν αυτούς τους δύο τύπους οντότητας είναι το **SSN** και το **PNUMBER** αντίστοιχα. Σύμφωνα με τον αλγόριθμο που περιγράψαμε στην προηγούμενη παράγραφο, δημιουργούμε ένα νέο πίνακα με το όνομα **WORKS_ON**, και προσθέτουμε σε αυτόν τα προαναφερόμενα πρωτεύοντα κλειδιά, αλλά με τα ονόματα **ESSN** και **PNO**. Επίσης από το μοντέλο οντοτήτων συσχετίσεων, διαπιστώνουμε πως ο τύπος συσχέτισης **WORKS_ON**, περιλαμβάνει και ένα πεδίο με το όνομα **HOURS** – το προσθέτουμε λοιπόν και αυτό. Το πρωτεύον κλειδί του νέου πίνακα **WORKS_ON** είναι **σύνθετο** και αποτελείται από το συνδυασμό των πεδίων **ESSN** και **PNO**. Έτσι, τελικά, ο πίνακας **WORKS_ON**, θα έχει την ακόλουθη δομή :

WORKS_ON {ESSN, PNO, HOURS}

ΒΗΜΑ 6 : Για κάθε πεδίο πολλαπλής τιμής (**multivalued attribute**), **A**, δημιουργούμε ένα καινούριο πίνακα, **R**, ο οποίος περιέχει το πεδίο **A**, καθώς και το **πρωτεύον κλειδί** του πίνακα που έχουμε δημιουργήσει στο **Βήμα 1**, για τον τύπο οντότητας που διαθέτει το πεδίο **A**. Εάν το πεδίο **A** είναι **σύνθετο πεδίο**, τότε τοποθετούμε στο νέο πίνακα **R**, όλα τα απλά πεδία από τα οποία αποτελείται. Το **πρωτεύον κλειδί** του πίνακα **R**, είναι **σύνθετο κλειδί**, και αποτελείται από το πεδίο **A**, και από το **ξένο κλειδί** που έχουμε προσθέσει στο νέο πίνακα, σύμφωνα με την παραπάνω διαδικασία.

Στο παράδειγμά μας, το μοναδικό πεδίο πολλαπλής τιμής, είναι το πεδίο **Locations** του τύπου οντότητας **DEPARTMENT**. Δημιουργούμε λοιπόν ένα πίνακα – στο παράδειγμά μας τον ονομάζουμε **DEPT_LOCATIONS** – ο οποίος περιέχει το πεδίο **Locations** (με το όνομα **DLOCATION**) και το πρωτεύον κλειδί του πίνακα **DEPARTMENT** που έχουμε δημιουργήσει στο **Βήμα 1** για τον ομώνυμο τύπο οντότητας (με το όνομα **DNUMBER**). Το πρωτεύον κλειδί του πίνακα **DEPT_LOCATIONS** είναι σύνθετο κλειδί και αποτελείται από το συνδυασμό των πεδίων **DNUMBER** + **DLOCATION**. Έτσι τελικά, ο πίνακας **DEPT_LOCATIONS** θα παρουσιάζει την ακόλουθη δομή :

DEPT_LOCATIONS {DNUMBER, DLOCATION}

Ξεκινώντας λοιπόν από το **μοντέλο οντοτήτων συσχετίσεων** (σελ. 27) και εφαρμόζοντας τα παραπάνω **έξι βήματα**, κατασκευάσαμε τελικά το **σχεσιακό σχήμα** για τη βάση δεδομένων της εταιρείας (σελ 35). Στην επόμενη σελίδα, παρουσιάζεται ξανά το ίδιο σχήμα, χωρίς τις συσχετίσεις των πεδίων όπως αυτές υπαγορεύονται από τον τρίτο κανόνα ακεραιότητας, αλλά με σαφή χαρακτηρισμό των πεδίων του κάθε πίνακα, από όπου μπορεί κανείς να καταλάβει, **ποια πεδία προήλθαν από άλλους πίνακες (foreign keys) καθώς και ποια πεδία έχουν προστεθεί, ως πεδία κάποιας συσχέτισης (relationship attributes)**.

EMPLOYEE

								f.k	f.k
FNAME	MINIT	LNAME	<u>SSN</u>	BDATE	ADDRESS	SEX	SALARY	SUPERSSN	DNO

DEPARTMENT

		f.k	rel.attr
DNAME	<u>DNUMBER</u>	MGRSSN	MGRSTARTDATE

DEPT_LOCATIONS

f.k
<u>DNUMBER</u>
<u>DLOCATION</u>

DEPENDENT

f.k				
<u>ESSN</u>	<u>DEPENDENT-NAME</u>	SEX	BDATE	RELATIONSHIP

PROJECT

			f.k
PNAME	<u>PNUMBER</u>	PLOCATION	DNUM

WORKS_ON

f.k	f.k	rel.attr
<u>ESSN</u>	<u>PNO</u>	HOURS

Σχήμα 10 : Το σχεσιακό σχήμα για τη βάση δεδομένων της εταιρείας Στο σχήμα αυτό, τα πεδία που φέρουν την ένδειξη f.k είναι ξένα κλειδιά (foreign keys) ενώ τα πεδία με την ένδειξη rel.attr είναι πεδία που έχουν προέλθει από τύπους συσχετίσεων. Τα πεδία αυτών των δύο κατηγοριών έχουν προστεθεί στους πίνακες της βάσης κατά τη διαδικασία μετατροπής του μοντέλου οντοτήτων συσχετίσεων σε σχεσιακό μοντέλο βάσεων δεδομένων.