

**ΟΙΚΟΝΟΜΙΚΟ  
ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΑΘΗΝΩΝ**



**ATHENS UNIVERSITY  
OF ECONOMICS  
AND BUSINESS**

# Συστήματα Διαχείρισης και Ανάλυσης Δεδομένων

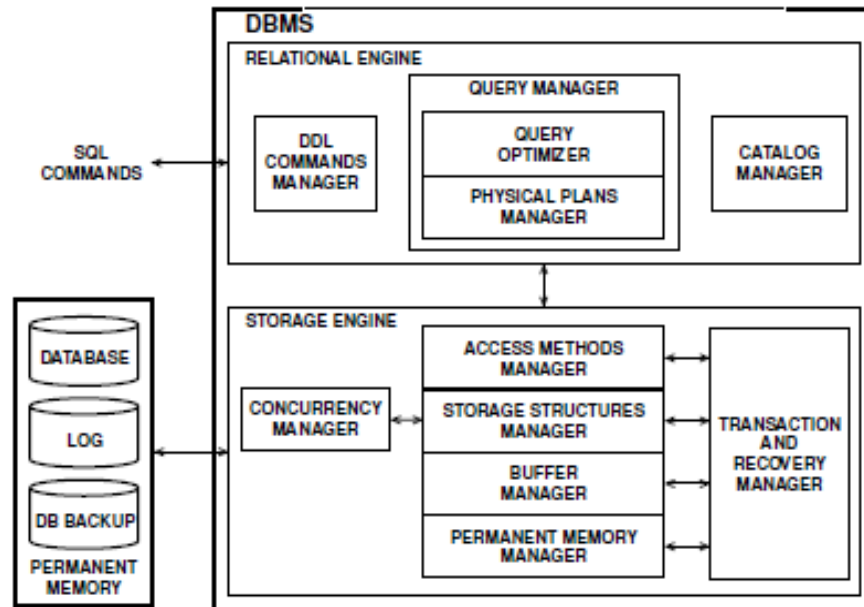
Διδάσκων Καθηγητής  
Ι. Κωτίδης

Φροντιστήριο 2

Καπέτης Χρυσόστομος  
mkar@aueb.gr

# Αρχιτεκτονική Συστημάτων Διαχείρισης Βάσεων Δεδομένων (DBMS)

- Ένα απλοποιημένο μοντέλο αρχιτεκτονικής ενός DBMS περιλαμβάνει τα ακόλουθα βασικά συστατικά (components).



Στην πραγματικότητα η λειτουργικότητα όλων αυτών των επιμέρους συστατικών δεν είναι τόσο ξεκάθαρη όσο φαίνεται στο διπλανό σχήμα.

Source: A. Albano, D. Colazzo, G. Ghelli, R. Orsini, DBMS Internal, 2015.

- Relational Engine: DBMS “Brain”**
- Storage Engine : DBMS “Muscle”**

# Relational Engine

- **Data Definition Language (DDL) Manager:** Επεξεργάζεται το σχήμα της βάσης.
- **Query Manager:** επεξεργάζεται το ερώτημα ενός χρήστη και το μετατρέπει σε μια ισοδύναμη αλλά πιο αποτελεσματική μορφή βρίσκοντας έτσι μια καλή στρατηγική για την εκτέλεσή του.
- **Catalogue Manager:** διαχειρίζεται τα μεταδεδομένα της βάσης.

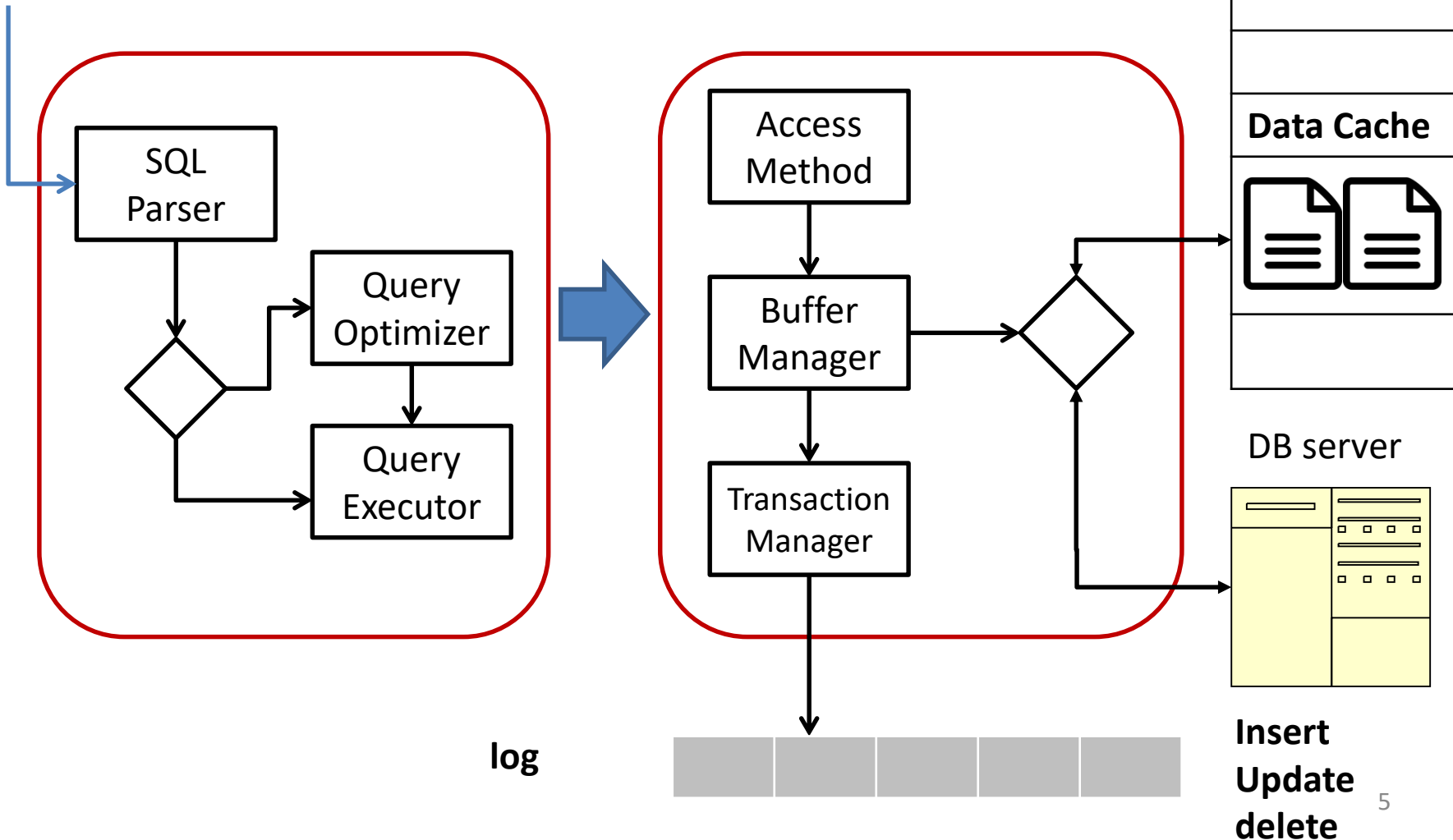
# Storage Engine

- **Permanent Memory Manager:** διαχειρίζεται τις σελίδες στον δίσκο.
- **Buffer Manager:** διαχειρίζεται την μεταφορά των σελίδων από τον δίσκο στην μνήμη και αντίστροφα.
- **Storage Structure Manager:** διαχειρίζεται τις δομές δεδομένων με στόχο την αποθήκευση και αποτελεσματική ανάκτηση των δεδομένων.
- **Access Methods Manager:** παρέχει την απαιτούμενη λειτουργικότητα για την δημιουργία και την διαγραφή βάσεων, αρχείων και δεικτών καθώς επίσης και τις μεθόδους πρόσβασης για σάρωση πινάκων και δεικτών.
- **Transaction and Recovery Manager:** διασφαλίζει την ακεραιότητα της βάσης δεδομένων σε περιπτώσεις αποτυχημένων συναλλαγών και αστοχιών του συστήματος
- **Concurrency Manager:** φροντίζει για την αποφυγή συγκρούσεων σε περιπτώσεις ταυτόχρονης πρόσβασης.



# SQL Server: Query Life Cycle

SELECT



# Μελέτη Περίπτωσης

Μία πολυεθνική εταιρεία διαθέτει ένα ηλεκτρονικό κατάστημα μέσω του οποίου οι καταναλωτές έχουν πρόσβαση σε έναν μεγάλο αριθμό προϊόντων οικιακού εξοπλισμού και εξοπλισμού γραφείου. Ο συνολικός αριθμός των διαθέσιμων προϊόντων ανέρχεται σε 15000.

Η εταιρεία θέλει να τηρεί στατιστικά στοιχεία σχετικά με την επισκεψιμότητα των προϊόντων της.

Το πλέον σημαντικό υπόδειγμα ερωτήματος που πρέπει να απαντηθεί είναι της μορφής:

- Ποιος είναι ο αριθμός των επισκέψεων του προϊόντος  $X$  για την περίοδο  $Y$ ;

## **Ζητείται:**

- Α. Να εκτιμηθεί ο αποθηκευτικός χώρος που απαιτείται για την τήρηση των απαιτούμενων στατιστικών στοιχείων ενός έτους.
- Β. Να υλοποιηθεί ένας πίνακας στον οποίο να καταχωρούνται τα στατιστικά στοιχεία. Η σχεδίαση του πίνακα να γίνει έτσι ώστε ερωτήματα της παραπάνω μορφής να εκτελούνται με τρόπο αποδοτικό.

# Λογικό και φυσικό Επίπεδο

- Το πρόβλημα φαντάζει απλό για κάποιον που διαθέτει βασικές γνώσεις των λογικών δομών αποθήκευσης και δεικτοδότησης (πίνακες και δείκτες) που διαθέτει ένα DBMS .
- Ωστόσο βασικές γνώσεις των δομών δεδομένων σε φυσικό επίπεδο (εγγραφές και σελίδες), οι οποίες συνθέτουν τις παραπάνω λογικές δομές, μας επιτρέπουν να πάρουμε καλύτερες αποφάσεις σχετικά με τις απαιτήσεις σε αποθηκευτικό χώρο και τις φυσικές δομές που απαιτούνται για αποδοτική ανάκτηση των πληροφοριών (βέλτιστη εκτέλεση ερωτήσεων).

## Εγγραφές (Records)

- Μία εγγραφή είναι η μικρότερη αποθηκευτική δομή ενός αρχείου στον SQL Server.
- Κάθε γραμμή ενός πίνακα αποθηκεύεται στον δίσκο με την μορφή μιας εγγραφής.
- Υπό την μορφή εγγραφών δεν αποθηκεύονται μόνο τα δεδομένα των πινάκων.
- Υπό την μορφή εγγραφών αποθηκεύονται επίσης οι δείκτες και τα μεταδεδομένα.



# Εγγραφές (Records)

- Μία εγγραφή αποτελείται από έναν αριθμό πεδίων σταθερού ή μεταβλητού μήκους.
- Υπάρχουν διάφοροι τρόποι αποθήκευσης των δεδομένων μιας εγγραφής.

A fixed-length record			
Attribute	Position	Value type	Value
Name	1	char(10)	Rossi
StudentCode	2	char(6)	456
City	3	char(2)	MI
BirthYear	4	int(2)	68

Total number of characters = 20

Attribute values are separated:

- a) by position                      Rossi\_\_\_\_\_456\_\_\_\_\_MI68
- b) with a separator                Rossi@456@MI@68
- c) with an index\*                  (1, 6, 9, 11) Rossi456MI68
- d) with labels\*\*                    (6, 1, Rossi)(4, 2, 456)(3, 3, MI)(3, 4, 68)

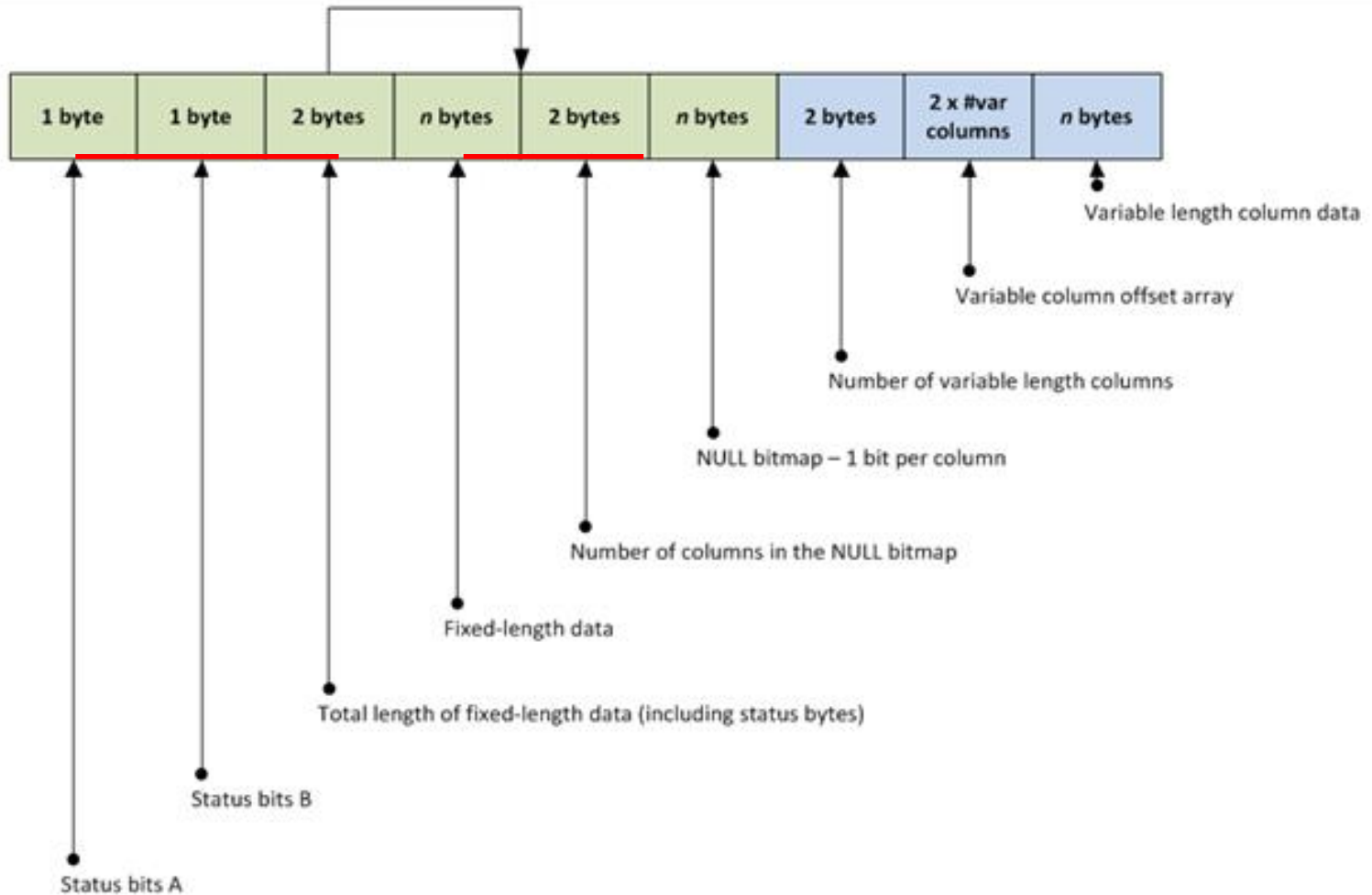
\* The index, placed at the beginning of each record, indicates the beginning of each attribute value.

\*\* Each attribute value starts with a counter that tells how many characters are used to code the position of the attribute and its value.

Source: A. Albano, D. Colazzo, G. Ghelli, R. Orsini, DBMS Internal, 2015.

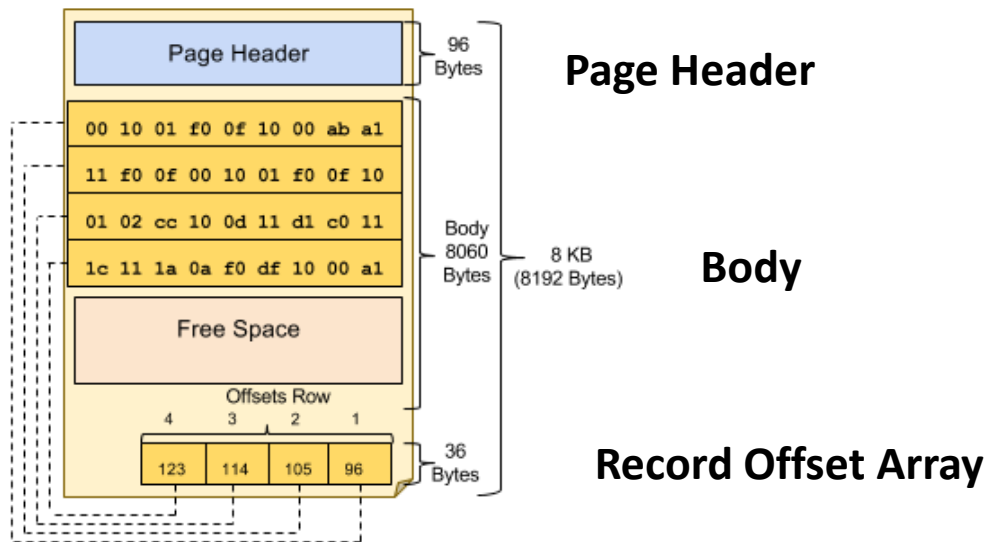
# SQL SERVER: Δομή Εγγραφής

- Στον SQL Server μια εγγραφή αποθηκεύεται σε μορφή **fixedvar**.

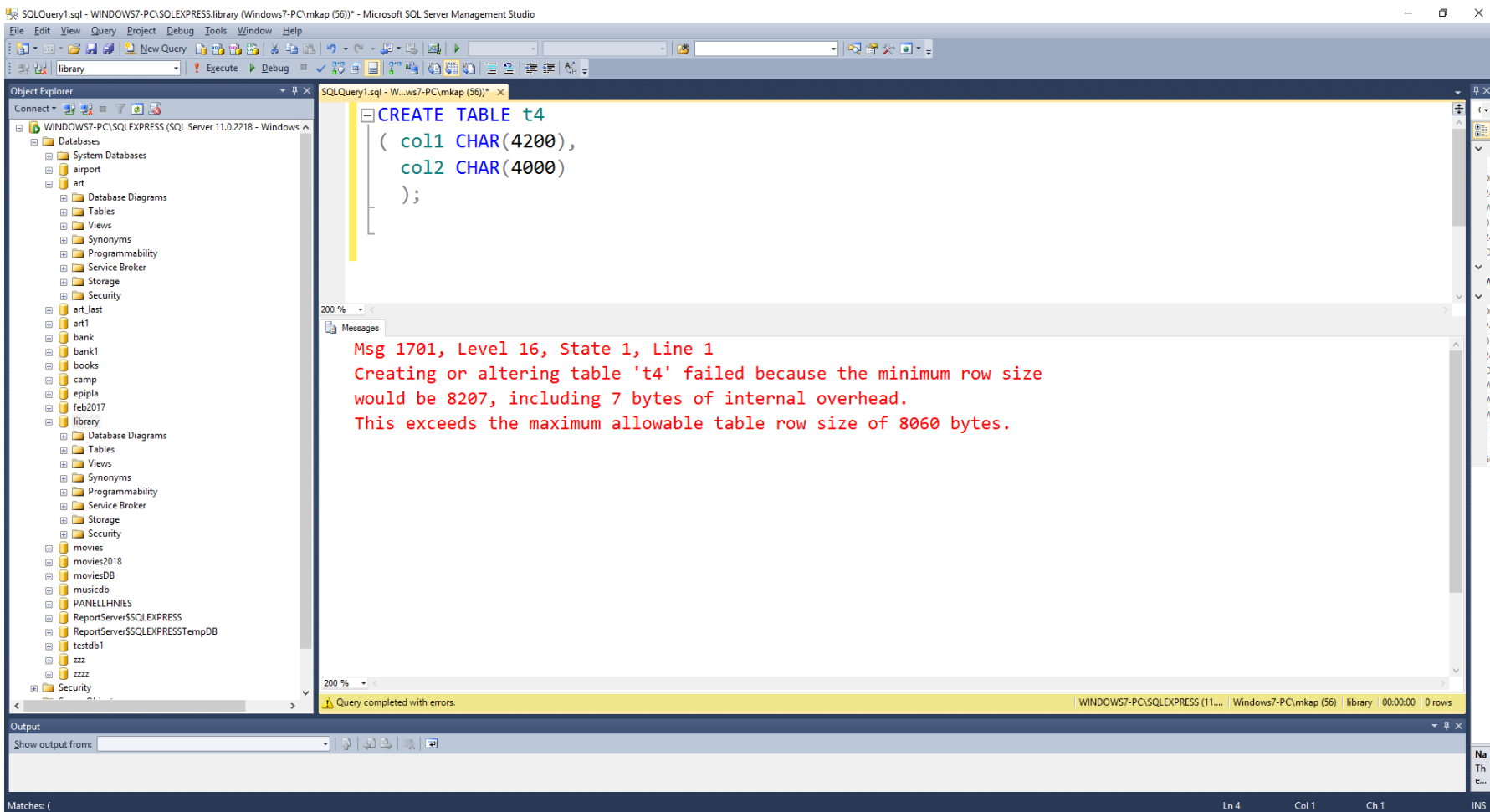


# SQL SERVER: Δομή Σελίδας

- Οι εγγραφές αποθηκεύονται σε σελίδες.
- Τα δεδομένα μεταφέρονται από τον δίσκο στην μνήμη και αντίστροφα υπό την μορφή σελίδων, τις οποίες διαχειρίζεται ο **Buffer Manager**.
- Η δομή μιας σελίδας στον SQL server είναι η ακόλουθη.



# ΠΡΟΣΟΧΗ: Το μέγεθος της εγγραφής δεν μπορεί να υπερβαίνει το μέγεθος της σελίδας.



The screenshot shows the Microsoft SQL Server Management Studio interface. The main window displays a SQL query: `CREATE TABLE t4 ( co11 CHAR(4200), co12 CHAR(4000) );`. The Object Explorer on the left shows a server instance with various databases, including 'library'. The Messages pane at the bottom shows an error message: `Msg 1701, Level 16, State 1, Line 1 Creating or altering table 't4' failed because the minimum row size would be 8207, including 7 bytes of internal overhead. This exceeds the maximum allowable table row size of 8060 bytes.` The status bar at the bottom indicates 'Query completed with errors.' and '0 rows'.

```
CREATE TABLE t4
(
  co11 CHAR(4200),
  co12 CHAR(4000)
);
```

Msg 1701, Level 16, State 1, Line 1  
Creating or altering table 't4' failed because the minimum row size  
would be 8207, including 7 bytes of internal overhead.  
This exceeds the maximum allowable table row size of 8060 bytes.

Query completed with errors. | WINDOW7-PC\SQLEXPRESS (11... | Windows7-PC\mkap (56) | library | 00:00:00 | 0 rows

# Στο Παράδειγμά μας

- Έστω ότι κάθε προϊόν προσδιορίζεται μοναδικά από έναν κωδικό (**ProductID**), ο οποίος είναι ακέραιο αριθμός.
- Ο αριθμός των επισκέψεων (**visitCount**) μπορεί επίσης να αναπαρασταθεί με ένα ακέραιο αριθμό.
- Για την ημερομηνία επίσκεψης (**visitDate**) μπορούμε να χρησιμοποιήσουμε τον τύπο **SMALLDATETIME**.
- Τέλος για την ώρα **visitHour** θα χρησιμοποιήσουμε τον τύπο **TINYINT**.
- Η δομή της εγγραφής μας θα είναι η εξής:

ProductID	INT	4 bytes
VisitDate	SMALLDATETIME	4 bytes
VisitHour	TINYINT	1 byte
VisitCount	INT	4 bytes
<b>TOTAL</b>		<b>13 bytes</b>

- Η εγγραφή όμως δεν περιέχει μόνο τα δεδομένα. Χρειαζόμαστε επίσης:
  - 2 **status** bytes
  - 2 **fixed length** bytes
  - 2 **null bitmap length indicator** bytes
  - 1 byte for **null bitmap**
  - 2 bytes in the **record offset array** in the end of the page

**ΣΥΝΟΛΙΚΟ ΜΕΓΕΘΟΣ ΕΓΓΡΑΦΗΣ=13+9=22 bytes**

Εφόσον το μέγεθος της σελίδας είναι **8096** bytes μπορούμε σε μια σελίδα να αποθηκεύσουμε **8096/22=368** εγγραφές.

# Δημιουργία Πίνακα Δίχως Δείκτη

Δημιουργούμε τον πίνακα:

```
CREATE TABLE StatsNoIndex
( ProductID INT NOT NULL,
  VisitDate SMALLDATETIME NOT NULL,
  VisitHour TINYINT NOT NULL,
  VisitCount INT NOT NULL
);
```

Φορτώνουμε 1.000.000 Εγγραφές:

```
Declare @productID INT
Declare @VisitDate SMALLDATETIME
Declare @VisitHour TINYINT
Declare @VisitCount INT
Declare @x INT
Set @x = 1

While @x <= 1000000
Begin
  SET @productID=(SELECT FLOOR (100000 + (115000-100000+1)*RAND()))
  SET @VisitDate=(SELECT DATEADD(DAY,ABS(CHECKSUM(NEWID()))%365,'2019-01-01'))
  SET @VisitHour=(SELECT FLOOR (1 + (24-1+1)*RAND()))
  SET @VisitCount=(SELECT FLOOR (1 + (10000-1+1)*RAND()))
  INSERT StatsNoIndex values (@productid, @visitDate, @visitHour, @VisitCount)
  SET @x=@x+1
End
```

# Η εντολή DBCC IND

## **DBCC IND (<database>, <object>, <IndexID>)**

Μας δείχνει ποιες σελίδες αντιστοιχούν σε έναν αντικείμενο (πίνακα) της βάσης.

- database: Το όνομα της βάσης
- object: Το όνομα του πίνακα
- IndexID: 0 , 1 , -1 (index type)

# Η εντολή DBCC PAGE

**DBCC PAGE (<database>, <FileID>, <PageID> <style>)**

Εμφανίζει τα περιεχόμενα μιας σελίδας .

- database: Το όνομα της βάσης
- FileID: Ο κωδικός του αρχείου
- PageID: Ο αριθμός της σελίδας
- style: Μορφή εμφάνισης των αποτελεσμάτων (0,1,2,3)

Η εντολή DBCC PAGE στέλνει τα αποτελέσματα σε κάποιο log αρχείο. Για να δούμε τα αποτελέσματα στην οθόνη πρέπει να ενεργοποιήσουμε το **flag3604** :

DBCC TRACEON(3604)

DBCC PAGE (eshop, 1, 1300, 3)

DBCC TRACEOFF(3604)



## Στατιστικά Στοιχεία (TIME)

### **SET STATISTICS TIME ON**

Εμφανίζει στατιστικά στοιχεία σχετικά με τον χρόνο εκτέλεσης μιας συναλλαγής.

### **SET STATISTICS TIME OFF**

Απενεργοποιεί την εμφάνιση στατιστικών στοιχείων

# Στατιστικά Στοιχεία (IO)

## **SET STATISTICS IO ON**

Εμφανίζει στατιστικά στοιχεία σχετικά με τις απαιτήσεις IO μιας συναλλαγής.

## **SET STATISTICS IO OFF**

Απενεργοποιεί την εμφάνιση των στατιστικών στοιχείων .

# Υπολογίσαμε Σωστά;

- Με βάση τους υπολογίσιμους μας για 100000 εγγραφές απαιτούνται  $100000/368=2717,39$  δηλαδή **2718** σελίδες δηλαδή  $2718 \times 8192 / 1024 / 1024 = \mathbf{21,23MB}$ .

DBCC IND (eshop, StatsNoIndex, 0)

	PageFID	PagePID	IAMFID	IAMPID	ObjectID	IndexID	PartitionNumber	PartitionID	iam_chain_type	PageType	IndexLevel	NextPageFID	NextPagePID	PrevPageFID
2840	1	9196	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
2841	1	9197	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
2842	1	9198	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
2843	1	9199	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
2844	1	9200	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
2845	1	9201	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
2846	1	9202	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
2847	1	9203	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
2848	1	9204	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
2849	1	9205	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
2850	1	9206	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
2851	1	9207	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
2852	1	9208	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
2853	1	9209	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
2854	1	9210	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
2855	1	9211	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
2856	1	9212	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
2857	1	9213	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
2858	1	9214	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
2859	1	9215	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0

Query executed successfully. MAKIS-PC\MSSQLSERVER01 (14... | MAKIS-PC\Makis (65) | CAMP | 00:00:02 | 2859 rows

# Εμφάνιση Σελίδων του Πίνακα: StatsNoIndex

SQLQuery2.sql - MAKIS-PC\MSSQLSERVER01.CAMP (MAKIS-PC\Makis (65))\* - Microsoft SQL Server Management Studio

File Edit View Query Project Debug ApexSQL Tools Window Help

CAMP Execute Debug

Object Explorer: MAKIS-PC > Databa > Sys > Dat > esh

SQLQuery2.sql - MA...KIS-PC\Makis (65)\* | SQLQuery1.sql - MA...KIS-PC\Makis (52)\*

DBCC IND (eshop, StatsNoIndex, 0)

150 %

PageFID	PagePID	IAMFID	IAMPID	ObjectID	IndexID	PartitionNum...	PartitionID	iam_chain_type	PageType	IndexLevel	NextPageFID	NextPagePID	PrevPageFID	PrevPagePID
1	73	NULL	NULL	1285579618	0	1	72057594043629568	In-row data	10	NULL	0	0	0	0
2	1	256	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
3	1	257	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
4	1	258	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
5	1	259	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
6	1	260	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
7	1	261	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
8	1	262	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
9	1	263	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
10	1	264	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
11	1	265	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
12	1	266	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
13	1	267	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
14	1	268	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
15	1	269	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
16	1	270	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
17	1	271	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0
18	1	280	1	73	1285579618	0	1	72057594043629568	In-row data	1	0	0	0	0

Query executed successfully. MAKIS-PC\MSSQLSERVER01 (14... | MAKIS-PC\Makis (65) | CAMP | 00:00:02 | 2859 rows

Ready Ln 1 Col 34 Ch 34 INS

# Περιεχόμενα Σελίδας Δεδομένων 256 (page 256)

SQLQuery2.sql - MAKIS-PC\MSSQLSERVER01.CAMP (MAKIS-PC\Makis (65))\* - Microsoft SQL Server Management Studio

File Edit View Query Project Debug ApexSQL Tools Window Help

Object Explorer: MAKIS-PC > Databases > esh > Data Files > CA

SQLQuery2.sql - MA...KIS-PC\Makis (65))\* SQLQuery1.sql - MA...KIS-PC\Makis (52)\*

```
DBCC TRACEON (3604)
DBCC PAGE (eshop, 1, 256,3)
DBCC TRACEOFF (3604)
```

100 %

Messages

PAGE HEADER:

Page @0x000001F2E92AA000

m_pageId = (1:256)	m_headerVersion = 1	m_type = 1
m_typeFlagBits = 0x0	m_level = 0	m_flagBits = 0x8200
m_objId (AllocUnitId.idObj) = 185	m_indexId (AllocUnitId.idInd) = 256	
Metadata: AllocUnitId = 72057594050052096		
Metadata: PartitionId = 72057594043629568	Metadata: IndexId = 0	
Metadata: ObjectId = 1285579618	m_prevPage = (0:0)	m_nextPage = (0:0)
pminlen = 17	m_slotCnt = 350	m_freeCnt = 396
m_freeData = 7096	m_reservedCnt = 0	m_lsn = (1237:18272:2)
m_xactReserved = 0	m_xdesId = (0:0)	m_ghostRecCnt = 0
m_tornBits = 1602250417	DB Frag ID = 1	

Allocation Status

GAM (1:2) = ALLOCATED	SGAM (1:3) = NOT ALLOCATED	PFS (1:1) = 0x44 ALLOCATED 100_PCT_FULL
DIFF (1:6) = CHANGED	ML (1:7) = NOT MIN_LOGGED	

Slot 0 Offset 0x60 Length 20

Record Type = PRIMARY\_RECORD Record Attributes = NULL\_BITMAP Record Size = 20

Memory Dump @0x0000004AA51FA060

```
0000000000000000: 10001100 0cb90100 00000bab 13cc1a00 00040000 .....?.....«.Ï.....
```

100 %

Query executed successfully. MAKIS-PC\MSSQLSERVER01 (14... MAKIS-PC\Makis (65) CAMP 00:00:01 0 rows

Ready Ln 2 Col 28 Ch 28 INS

PAGE HEADER:

Page @0x000001F2E92AA000

```
m_pageId = (1:256) m_headerVersion = 1 m_type = 1
m_typeFlagBits = 0x0 m_level = 0 m_flagBits = 0x8200
m_objId (AllocUnitId.idObj) = 185 m_indexId (AllocUnitId.idInd) = 256
Metadata: AllocUnitId = 72057594050052096
Metadata: PartitionId = 72057594043629568 Metadata: IndexId = 0
Metadata: ObjectId = 1285579618 m_prevPage = (0:0) m_nextPage = (0:0)
pminlen = 17 m_slotCnt = 350 m_freeCnt = 396
m_freeData = 7096 m_reservedCnt = 0 m_lsn = (1237:18272:2)
m_xactReserved = 0 m_xdesId = (0:0) m_ghostRecCnt = 0
m_tornBits = 1602250417 DB Frag ID = 1
```

Allocation Status

```
GAM (1:2) = ALLOCATED SGAM (1:3) = NOT ALLOCATED PFS (1:1) = 0x44
ALLOCATED 100_PCT_FULL
DIFF (1:6) = CHANGED ML (1:7) = NOT MIN_LOGGED
```

Slot 0 Offset 0x60 Length 20

```
Record Type = PRIMARY_RECORD Record Attributes = NULL_BITMAP Record Size = 20
```

Memory Dump @0x0000004AA51FA060

```
0000000000000000: 10001100 0cb90100 00000bab 13cc1a00 00040000 .....1.....«.Ì.....
```

```
Slot 0 Column 1 Offset 0x4 Length 4 Length (physical) 4
ProductID = 112908
```

```
Slot 0 Column 2 Offset 0x8 Length 4 Length (physical) 4
```

```
VisitDate = 2019-11-20 00:00:00.000
```

```
Slot 0 Column 3 Offset 0xc Length 1 Length (physical) 1
```

```
VIsitHour = 19
```

```
Slot 0 Column 4 Offset 0xd Length 4 Length (physical) 4
```

```
VisitCount = 6860
```

Slot 1 Offset 0x74 Length 20

```
Record Type = PRIMARY_RECORD Record Attributes = NULL_BITMAP Record Size = 20
```

## Στο παράδειγμά μας

- Υποθέτοντας ότι κάθε προϊόν δέχεται τουλάχιστον μία επίσκεψη ανά ώρα σε κάθε εικοσιτετράωρο (24/7), θα καταχωρηθούν στον πίνακά μας:

$$365 * 24 * 15000 = 131400000.$$

- Δηλαδή θα χρειαστούμε  $131400000/368=357065,22 = 358066$  σελίδες.

- Που σημαίνει  $358066 * 8KB = 2,9 GB$

# Δημιουργία Πίνακα με Δείκτη

Δημιουργούμε έναν νέο πίνακα StatsWithIndex:

```
CREATE TABLE StatsWithIndex
( ProductID INT NOT NULL,
  VisitDate SMALLDATETIME NOT NULL,
  VisitHour TINYINT NOT NULL,
  VisitCount INT NOT NULL
);
```

Δημιουργούμε τον δείκτη:

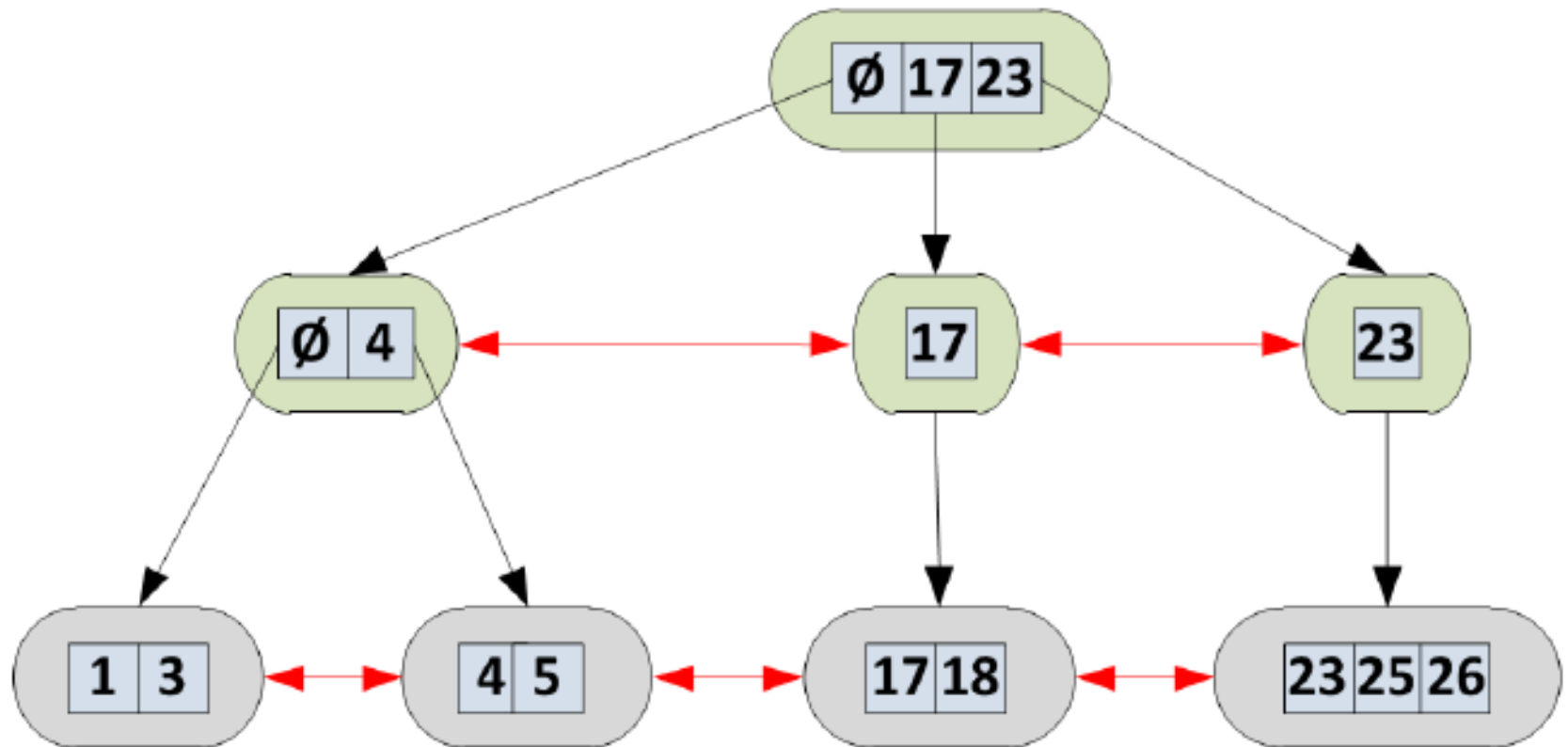
```
CREATE CLUSTERED INDEX CI1 ON StatsWithIndex(ProductId, VisitDate);
```

Φορτώνουμε τις ίδιες εγγραφές:

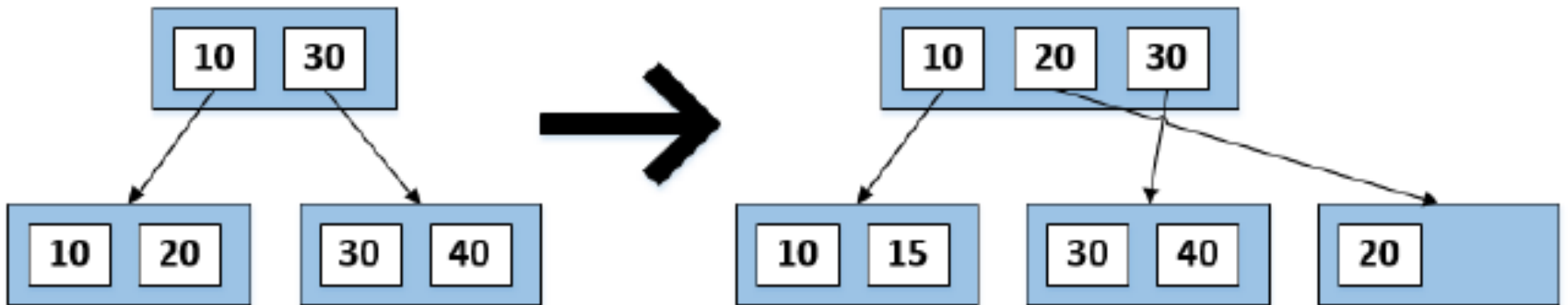
```
INSERT StatsWithIndex
  SELECT ProductID, VisitDate, VisitHour, VisitCount FROM StatsNoIndex
  ORDER BY ProductID, VisitDate
```



# B-trees



# Page Split



# Εμφάνιση Σελίδων του Πίνακα: StatsWithIndex

SQLQuery1.sql - W...ws7-PC\mkap (76)\*

DBCC IND (eshop, StatsWithIndex, 1)

150 %

Results Messages

	PageFID	PagePID	IAMFID	IAMPID	ObjectID	IndexID	PartitionNumber	PartitionID	iam_chain_type	PageType	IndexLevel	NextPageFID	NextPagePID	PrevPageFID	PrevPagePID
503	1	3803	1	150	917578307	1	1	72057594043301888	In-row data	1	0	1	3804	1	3802
504	1	3804	1	150	917578307	1	1	72057594043301888	In-row data	1	0	1	3805	1	3803
505	1	3805	1	150	917578307	1	1	72057594043301888	In-row data	1	0	1	3806	1	3804
506	1	3806	1	150	917578307	1	1	72057594043301888	In-row data	1	0	1	3807	1	3805
507	1	3807	1	150	917578307	1	1	72057594043301888	In-row data	1	0	1	3808	1	3806
508	1	3808	1	150	917578307	1	1	72057594043301888	In-row data	1	0	1	3809	1	3807
509	1	3809	1	150	917578307	1	1	72057594043301888	In-row data	1	0	1	3810	1	3808
510	1	3810	1	150	917578307	1	1	72057594043301888	In-row data	1	0	1	3811	1	3809
511	1	3811	1	150	917578307	1	1	72057594043301888	In-row data	1	0	1	3812	1	3810
512	1	3812	1	150	917578307	1	1	72057594043301888	In-row data	1	0	1	3813	1	3811
513	1	3813	1	150	917578307	1	1	72057594043301888	In-row data	1	0	1	3814	1	3812
514	1	3814	1	150	917578307	1	1	72057594043301888	In-row data	1	0	1	3815	1	3813
515	1	3815	1	150	917578307	1	1	72057594043301888	In-row data	1	0	1	3832	1	3814
516	1	3816	1	150	917578307	1	1	72057594043301888	In-row data	2	2	0	0	0	0
517	1	3817	1	150	917578307	1	1	72057594043301888	In-row data	2	1	1	3818	1	3296
518	1	3818	1	150	917578307	1	1	72057594043301888	In-row data	2	1	1	3819	1	3817
519	1	3819	1	150	917578307	1	1	72057594043301888	In-row data	2	1	1	3820	1	3818
520	1	3820	1	150	917578307	1	1	72057594043301888	In-row data	2	1	1	3821	1	3819
521	1	3821	1	150	917578307	1	1	72057594043301888	In-row data	2	1	1	3822	1	3820
522	1	3822	1	150	917578307	1	1	72057594043301888	In-row data	2	1	1	3823	1	3821
523	1	3823	1	150	917578307	1	1	72057594043301888	In-row data	2	1	1	3824	1	3822
524	1	3824	1	150	917578307	1	1	72057594043301888	In-row data	2	1	1	3825	1	3823
525	1	3825	1	150	917578307	1	1	72057594043301888	In-row data	2	1	1	3826	1	3824
526	1	3826	1	150	917578307	1	1	72057594043301888	In-row data	2	1	1	3827	1	3825

# Περιεχόμενα Σελίδας Δείκτη 3816 (page 3816)

SQLQuery1.sql - W...ws7-PC\mkap (76))\*

DBCC TRACEON (3604)  
DBCC PAGE (eshop, 1, 3816, 3)  
DBCC TRACEON (3604)

150 %

Results Messages

	FileId	PageId	Row	Level	ChildFileId	ChildPageId	ProductID (key)	VisitDate (key)	UNIQUIFIER (key)	KeyHashValue	Row Size
1	1	3816	0	2	1	3296	NULL	NULL	NULL	NULL	15
2	1	3816	1	2	1	3817	101070	2019-04-14 00:00:00	0	NULL	15
3	1	3816	2	2	1	3818	102145	2019-11-27 00:00:00	0	NULL	15
4	1	3816	3	2	1	3819	103233	2019-07-20 00:00:00	1	NULL	23
5	1	3816	4	2	1	3820	104298	2019-06-05 00:00:00	1	NULL	23
6	1	3816	5	2	1	3821	105382	2019-03-06 00:00:00	0	NULL	15
7	1	3816	6	2	1	3822	106452	2019-08-29 00:00:00	0	NULL	15
8	1	3816	7	2	1	3823	107527	2019-01-04 00:00:00	0	NULL	15
9	1	3816	8	2	1	3824	108620	2019-09-25 00:00:00	0	NULL	15
10	1	3816	9	2	1	3825	109697	2019-04-28 00:00:00	0	NULL	15
11	1	3816	10	2	1	3826	110789	2019-01-29 00:00:00	0	NULL	15
12	1	3816	11	2	1	3827	111887	2019-12-31 00:00:00	1	NULL	23
13	1	3816	12	2	1	3828	112977	2019-08-10 00:00:00	0	NULL	15

# Περιεχόμενα Σελίδας Δείκτη 3818 (page 3818)

SQLQuery1.sql - W...ws7-PC\mkap (76)\*

- DBCC TRACEON (3604)
- DBCC PAGE (eshop, 1, 3818, 3)
- DBCC TRACEON (3604)

150 %

Results Messages Client Statistics

	FileId	PageId	Row	Level	ChildFileId	ChildPageId	ProductID (key)	VisitDate (key)	UNIQUIFIER (key)	KeyHashValue	Row Size
1	1	3818	0	1	1	3706	102145	2019-11-27 00:00:00	0	NULL	15
2	1	3818	1	1	1	3707	102151	2019-03-28 00:00:00	0	NULL	15
3	1	3818	2	1	1	3708	102156	2019-04-20 00:00:00	0	NULL	15
4	1	3818	3	1	1	3709	102161	2019-08-01 00:00:00	0	NULL	15
5	1	3818	4	1	1	3710	102166	2019-10-21 00:00:00	0	NULL	15
6	1	3818	5	1	1	3711	102171	2019-12-08 00:00:00	0	NULL	15
7	1	3818	6	1	1	3712	102177	2019-03-03 00:00:00	0	NULL	15
8	1	3818	7	1	1	3713	102182	2019-10-30 00:00:00	0	NULL	15
9	1	3818	8	1	1	3714	102188	2019-01-27 00:00:00	0	NULL	15
10	1	3818	9	1	1	3715	102193	2019-09-19 00:00:00	0	NULL	15
11	1	3818	10	1	1	3716	102199	2019-04-10 00:00:00	0	NULL	15
12	1	3818	11	1	1	3717	102204	2019-08-31 00:00:00	0	NULL	15
13	1	3818	12	1	1	3718	102209	2019-08-30 00:00:00	0	NULL	15
14	1	3818	13	1	1	3719	102214	2019-12-10 00:00:00	1	NULL	23

# Περιεχόμενα Σελίδας 3707 (page 3707)

```
SQLQuery1.sql - W...ws7-PC\mkap (76)*  - X
└─ DBCC TRACEON (3604)
  └─ DBCC PAGE (eshop, 1, 3707, 3)
    └─ DBCC TRACEON (3604)

150 %
Messages

Record Type = PRIMARY_RECORD      Record Attributes = NULL_BITMAP      Record Size = 20

Memory Dump @0x000000719FCF8060

0000000000000000:  10001100 078f0100 00001eaa 02da2600 00050000  .....ű&.....

Slot 0 Column 0 Offset 0x0 Length 4 Length (physical) 0

UNIQUIFIER = 0

Slot 0 Column 1 Offset 0x4 Length 4 Length (physical) 4

ProductID = 102151

Slot 0 Column 2 Offset 0x8 Length 4 Length (physical) 4

VisitDate = 2019-03-28 00:00:00.000

Slot 0 Column 3 Offset 0xc Length 1 Length (physical) 1

VisitHour = 2

Slot 0 Column 4 Offset 0xd Length 4 Length (physical) 4

VisitCount = 9946

Slot 0 Offset 0x0 Length 0 Length (physical) 0

KeyHashValue = (4dcd17265ac4)
Slot 1 Offset 0x74 Length 20
```

# Στατιστικά Στοιχεία (time)

The screenshot displays the Microsoft SQL Server Management Studio interface. The main window shows a SQL query being executed in the 'SQLQuery1.sql' file. The query is as follows:

```
DBCC DROPCLEANBUFFERS
SET STATISTICS TIME ON

SELECT COUNT(*) FROM StatsNoIndex
SELECT COUNT(*) FROM StatsWithIndex

SET STATISTICS TIME OFF
```

The 'Results' pane shows the output of the query execution:

```
DBCC execution completed. If DBCC printed error messages, contact your system administrator.

(1 row affected)

SQL Server Execution Times:
  CPU time = 375 ms,  elapsed time = 543 ms.

(1 row affected)

SQL Server Execution Times:
  CPU time = 437 ms,  elapsed time = 424 ms.
```

The status bar at the bottom indicates 'Query executed successfully.' and shows the server name 'MAKIS-PC\MSSQLSERVER01', the instance name 'MAKIS-PC\Makis (51)', the database 'eshop', and the execution time '00:00:01' with '2 rows' returned.

# Στατιστικά Στοιχεία (IO)

The screenshot displays the Microsoft SQL Server Management Studio interface. The main window shows a query window with the following SQL code:

```
DBCC DROPLEANBUFFERS
SET STATISTICS IO ON

SELECT COUNT(*) FROM StatsNoIndex
SELECT COUNT(*) FROM StatsWithIndex

SET STATISTICS IO OFF
```

The Results pane shows the output of the query:

```
DBCC execution completed. If DBCC printed error messages, contact your system administrator.

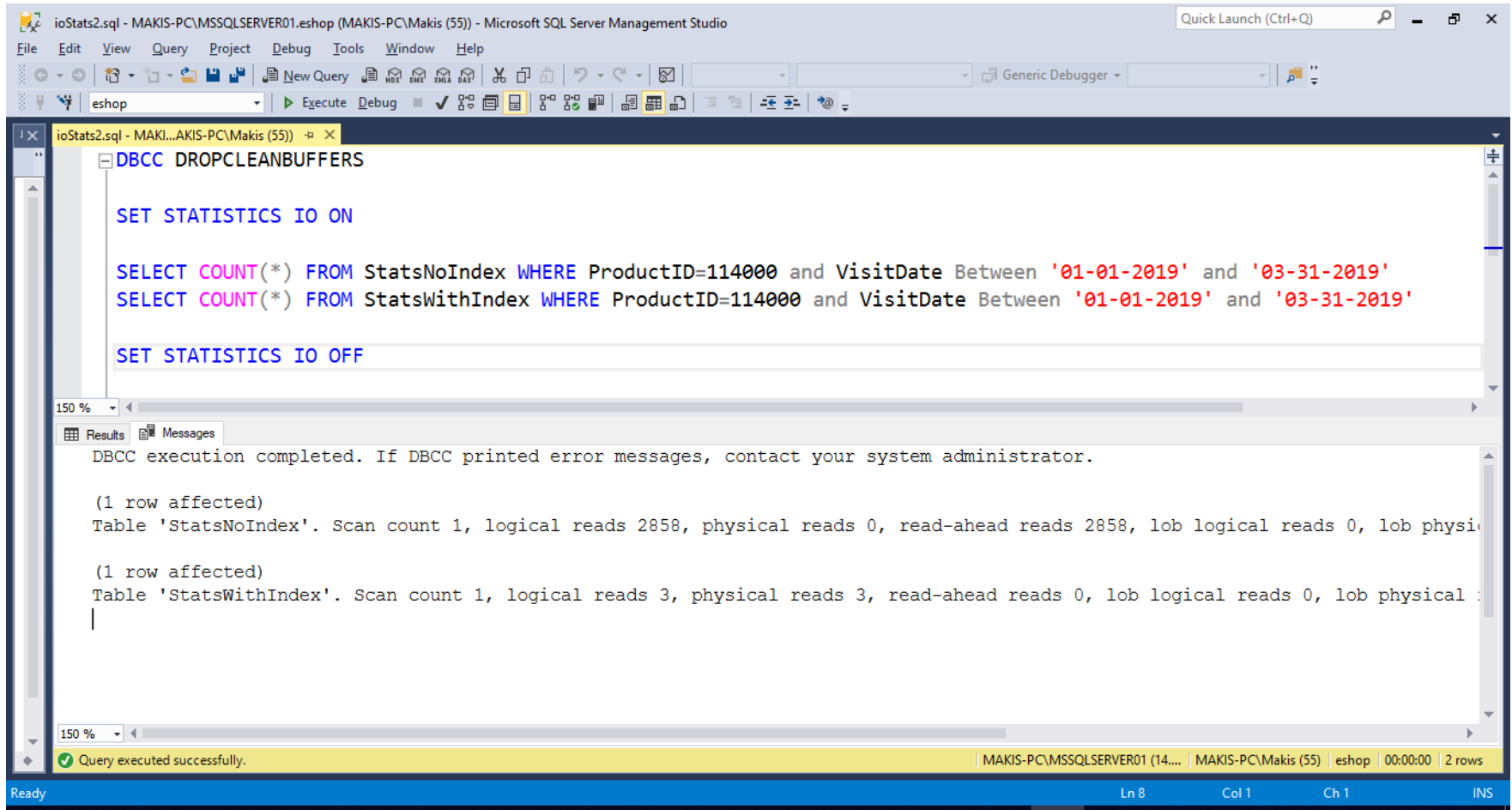
(1 row affected)
Table 'StatsNoIndex'. Scan count 1, logical reads 2858, physical reads 0, read-ahead reads 2857, lob logical reads 0, lob physical reads 0, lob read-ahead reads 0

(1 row affected)
Table 'StatsWithIndex'. Scan count 1, logical reads 2740, physical reads 0, read-ahead reads 0, lob logical reads 0, lob physical reads 0, lob read-ahead reads 0
```

The status bar at the bottom indicates: Query executed successfully. The status bar also shows the server name (MAKIS-PC\MSSQLSERVER01), the instance name (MAKIS-PC\Makis (54)), the database name (eshop), the execution time (00:00:00), and the number of rows affected (2 rows).



# Στατιστικά Στοιχεία (IO)



The screenshot displays the Microsoft SQL Server Management Studio interface. The main window shows a query editor with the following SQL script:

```
DBCC DROPCLEANBUFFERS

SET STATISTICS IO ON

SELECT COUNT(*) FROM StatsNoIndex WHERE ProductID=114000 and VisitDate Between '01-01-2019' and '03-31-2019'
SELECT COUNT(*) FROM StatsWithIndex WHERE ProductID=114000 and VisitDate Between '01-01-2019' and '03-31-2019'

SET STATISTICS IO OFF
```

Below the query editor, the Results pane shows the output of the query execution:

```
DBCC execution completed. If DBCC printed error messages, contact your system administrator.

(1 row affected)
Table 'StatsNoIndex'. Scan count 1, logical reads 2858, physical reads 0, read-ahead reads 2858, lob logical reads 0, lob physio

(1 row affected)
Table 'StatsWithIndex'. Scan count 1, logical reads 3, physical reads 3, read-ahead reads 0, lob logical reads 0, lob physical :
```

The status bar at the bottom indicates that the query was executed successfully, with a duration of 00:00:00 and 2 rows affected.