How Achaeans Would Construct Columns in Troy

Alekh Jindal, Felix Martin Schuhknecht, Jens Dittrich, Karen Khachatryan, Alexander Bunte



UNIVERSITÄT DES SAARLANDES



Number of Visas Received



Health Level 5 days before CIDR



Average Number of Slides per 20min talk



Number of Slides Actually Prepared 150 100 50 0 Alekh Jens



What is the problem?



Column-stores



OLTP



OLAP





OLTP



Can we do efficient OLAP in Row-stores?

Any solutions out there?



C-Tables * Application Query Processor

Relations

Physical Representation

File I	File 2	File 3	••••	File n
--------	--------	--------	------	--------

* Nicolas Bruno. Teaching an Old Elephant New Tricks. CIDR 2009

Relation

Customer			
name	phone	market_segment	
smith	2134	automobile	
john	3425	household	
kim	6756	furniture	
joe	9878	building	
mark	4312	building	
steve	2435	automobile	
jim	5766	household	
ian	8789	household	

* Nicolas Bruno. Teaching an Old Elephant New Tricks. CIDR 2009

Sorted Relation

Customer			
name	phone	market_segment	
smith	2134	automobile	
steve	2435	automobile	
mark	4312	building	
joe	9878	building	
kim	6756	furniture	
john	3425	household	
jim	5766	household	
ian	8789	household	

Physical Table

T_market_segment		
f	V	c
1	automobile	2
3	building	2
5	furniture	1
6	household	3

Sorted Relation

Customer			
name	phone	market_segment	
smith	2134	automobile	
steve	2435	automobile	
mark	4312	building	
joe	9878	building	
kim	6756	furniture	
john	3425	household	
jim	5766	household	
ian	8789	household	

Physical Table

T_market_segment		
f	V	c
1	automobile	2
3	building	2
5	furniture	1
6	household	3

* Nicolas Bruno. Teaching an Old Elephant New Tricks. CIDR 2009

Sorted Relation

Customer			
name	phone	market_segment	
smith	2134	automobile	
steve	2435	automobile	
mark	4312	building	
joe	9878	building	
kim	6756	furniture	
john	3425	household	
jim	5766	household	
ian	8789	household	

Physical Table

T_market_segment		
f	V	c
1	automobile	2
3	building	2
5	furniture	1
6	household	3

Sorted Relation

Customer			
name	phone	market_segment	
smith	2134	automobile	
steve	2435	automobile	
mark	4312	building	
joe	9878	building	
kim	6756	furniture	
john	3425	household	
jim	5766	household	
ian	8789	household	

Physical Table

T_market_segment		
f	V	c
1	automobile	2
3	building	2
5	furniture	1
6	household	3

Sorted Relation

Customer			
name	phone	market_segment	
smith	2134	automobile	
steve	2435	automobile	
mark	4312	building	
joe	9878	building	
kim	6756	furniture	
john	3425	household	
jim	5766	household	
ian	8789	household	

Physical Table

T_market_segment		
f	V	С
1	automobile	2
3	building	2
5	furniture	1
6	household	3

	T_j	phone	I_T	name
	f	V	f	V
	1	2134	1	smith
	2	2435	2	steve
X	3	4312	3	mark
	4	9878	4	joe
	5	6756	5	kim
	6	3425	6	john
	7	5766	7	jim
	8	8789	8	ian

JOINS !

* Nicolas Bruno. Teaching an Old Elephant New Tricks. CIDR 2009



CTables:	0.82199066
RowTime:	30.0123393



* Nicolas Bruno. Teaching an Old Elephant New Tricks. CIDR 2009







* **C-Tables**[†] **Application Query Processor Relations Physical Representation** File I File 2 File 3 File n ••••

User

Database





Relation

Customer				
name	phone	market_segment		
smith	2134	automobile		
john	3425	household		
kim	6756	furniture		
joe	9878	building		
mark	4312	building		
steve	2435	automobile		
jim	5766	household		
ian	8789	household		

Physical Table

Customer_trojan				
segment_ID	attribute_ID	blob_data		
1	name	smith, john, kim, joe		
1	phone	2134, 3425, 6756, 9878		
1	market_segment	automobile, household, furniture, building		
2	name	mark, steve, jim, ian		
2	phone	4312, 2435, 5766, 8789		
2	market_segment	building, automobile, household, household		

segment size = 4

Relation

Customer				
name	phone	market_segment		
smith	2134	automobile		
john	3425	household		
kim	6756	furniture		
joe	9878	building		
mark	4312	building		
steve	2435	automobile		
jim	5766	household		
ian	8789	household		

Physical Table

Customer_trojan				
segment_ID	attribute_ID	blob_data		
1	name	smith, john, kim, joe		
1	phone	2134, 3425, 6756, 9878		
1	market_segment	automobile, household, furniture, building		
2	name	mark, steve, jim, ian		
2	phone	4312, 2435, 5766, 8789		
2	market_segment	building, automobile, household, household		

segment size = 4

Relation

Customer				
name	phone	market_segment		
smith	2134	automobile		
john	3425	household		
kim	6756	furniture		
joe	9878	building		
mark	4312	building		
steve	2435	automobile		
jim	5766	household		
ian	8789	household		

Physical Table

Customer_trojan				
segment_ID	attribute_ID	blob_data		
1	name	smith, john, kim, joe		
1	phone	2134, 3425, 6756, 9878		
1	market_segment	automobile, household, furniture, building		
2	name	mark, steve, jim, ian		
2	phone	4312, 2435, 5766, 8789		
2	market_segment	building, automobile, household, household		

segment size = 4





DEEP CHANGES !





LONG TIME !





SOURCE CODE !





What do we propose?



Add	lication
· · P P	

Database

Query Processor					
Relations					
UDF Storage Layer					
Physical Representation					
File I	File 2	File 3	••••	File n	

Relation

Customer				
name	phone	market_segment		
smith	2134	automobile		
john	3425	household		
kim	6756	furniture		
joe	9878	building		
mark	4312	building		
steve	2435	automobile		
jim	5766	household		
ian	8789	household		

Physical Table

Customer_trojan			
segment_ID	attribute_ID	blob_data	
1	name	smith, john, kim, joe	
1	phone	2134, 3425, 6756, 9878	
1	market_segment	automobile, household, furniture, building	
2	name	mark, steve, jim, ian	
2	phone	4312, 2435, 5766, 8789	
2	market_segment	building, automobile, household, household	

Relation



Physical Table

Customer_trojan			
segment_ID	attribute_ID	blob_data	
1	name	smith, john, kim, joe	
1	phone	2134, 3425, 6756, 9878	
1	market_segment	automobile, household, furniture, building	
2	name	mark, steve, jim, ian	
2	phone	4312, 2435, 5766, 8789	
2	market_segment	building, automobile, household, household	

Relation



. Physical Table

Customer_trojan			
segment_ID	attribute_ID	blob_data	
1	name	smith, john, kim, joe	
1	phone	2134, 3425, 6756, 9878	
1	market_segment	automobile, household, furniture, building	
2	name	mark, steve, jim, ian	
2	phone	4312, 2435, 5766, 8789	
2	market_segment	building, automobile, household, household	
Result



lineitem









Trojan Columns

Application

Query Processor Relations UDF Storage Layer

Physical Representation

File I File 2 File 3 File n

Plug-and-play

User

Database



User

Database

Trojan Columns

Application

Query Processor Relations UDF Storage Layer **Physical Representation** File I File 2 File 3 File n

••••

Quick Deployment



Trojan Columns



Physical Representation

File I	File 2	File 3	••••	File n
--------	--------	--------	------	--------

Closed-source

User

Database



Trojan Columns

User

Database



Will this work?

Experimental Setup



Commercial closed-source Row-store (Standard Row)



Trojan Columns in commercial closed-source Row-store (Trojan Columns)



Three variants of TPC-H benchmark:

- I. simplified queries , simplified dataset
- 2. simplified queries , original dataset
- 3. original queries , original dataset

C	supplier nation:	10.144428 5.5 9 04746:	*** *** C:		
				\neg	

	Standard Row	Trojan Columns	Trojan Columns (SP)	Standard Row	Trojar
Q1	76.730296	19.293983	24.208052774	230.19089	57.8
Q6	77.589034	8.6532381	11.235220175	232.7671	25.9
Q12	92.486038	37.331905	40.598335758	277.45811	111.
Q14	81.207649	30.788114	59.597473787	243.62295	92.3
Q3	111.88261	809.38127		335.64782	2428
Q5	99.729039	169.34457		299.18712	508.
Q10	110.93664	119.46429		332.80993	358.
Q19	79.140857	43.115296		237.42257	129.

372.41696to7448325790r9807r3742d322138922491.637

	supplier	10.144428	***					
S	implified region	5.8037666	*** e\$,	Simp	lified	Dat	aset	*

		5x					
		\frown					
		Standa	rd Row	Trojan Columns	Trojan Columns (SP)	Standard Row	Trojar
Q1		76.73	0296	19.293983	24.208052774	230.19089	57.8
Q6		77.58	9034	8.6532381	11.235220175	232.7671	25.9
Q1:	2	92.48	6038	37.331905	40.598335758	277.45811	111.
Q1	4	81.20	7649	30.788114	59.597473787	243.62295	92.3
Q3		111.8	8261	809.38127		335.64782	2428
Q5		99.72	9039	169.34457		299.18712	508.
Q10	1	110.9	3664	119.46429		332.80993	358.
Q19	9	79.14	0857	43.115296		237.42257	129.

372.41696to7448325790r9807r3742d322138922491.637

en Ø 375

Simplified Queries, Original Dataset

V	Trojan Columns	Trojan Columns (O)	Trojan Columns (SP)	Factor	Factor
9	57.881948		72.62415832	3.977	3.17
	25.959714		33.70566052	8.966	6.906
1	111.99572		121.7950073	2.477	2.278
5	92.364343		178.7924214	2.638	1.363
2	2428.1438			0.138	
2	508.0337			0.589	
3	358.39288			0.929	
7	129.34589		227.3616356	1.836	
4					

10

Query Time (sec)

2

ong 375

Simplified Queries, Original Dataset

V	Trojan Columns	Trojan Columns (O)	Trojan Columns (SP)	Factor	Factor
9	57.881948		72.62415832	3.977	3.17
	25.959714		33.70566052	8.966	6.906
1	111.99572		121.7950073	2.477	2.278
5	92.364343		178.7924214	2.638	1.363
2	2428.1438			0.138	
2	508.0337			0.589	
3	358.39288			0.929	
7	129.34589		227.3616356	1.836	
4					

10

7

Query Time (sec)

2

42

Original Queries, Original Dataset * The "Good Queries"



Original Queries, Original Dataset * The "Good Queries"



What are the trade-offs?

292.16139 292.9931 241.52515 200.74212 Original Queries, Original Dataset *

The "Bad Queries"



		1E-06	1E-05	1E-04	1E-03	1E-02	1E-01	1E+00
#	1	17.43	17.61	16.61	13.57	4.39	0.57	0.06
t ref	3	12.86	13.57	13.22	11.03	4.16	0.56	0.06
ere	5	10.96	10.94	10.55	9.27	3.75	0.57	0.06
nce	7	7.00	6.96	6.80	6.23	3.17	0.56	0.06
d a	9	6.39	6.38	6.25	5.79	3.11	0.54	0.06
ttrik	11	5.79	5.82	5.75	5.24	2.87	0.56	0.06
oute	13	5.00	5.00	4.94	4.61	2.70	0.57	0.06
I) se	15	4.64	4.62	4.55	4.27	2.57	0.55	0.06
(-	16	2.13	2.13	2.11	2.06	1.55	0.47	0.06
	16	2 1 2	9 1 2	2 1 1	2.06	1 55	0 47	

selectivity (fraction of tuples accessed)

152.9599 153.2835 152.7586 152.8184 152.7484 qs4 qs3 qs2 qs1 qs0 IIIIPI Over NOver NOve-Store

Not Affected

		1E-06	1E-05	1E-04	1E-03	1E-02	1E-01	1E+00
Ŧ	1	17.43	17.61	16.61	13.57	4.39	0.57	0.06
Tef	3	12.86	13.57	13.22	11.03	4.16	0.56	0.06
ere	5	10.96	10.94	10.55	9.27	3.75	0.57	0.06
nce	7	7.00	6.96	6.80	6.23	3.17	0.56	0.06
d a	9	6.39	6.38	6.25	5.79	3.11	0.54	0.06
ttrik	11	5.79	5.82	5.75	5.24	2.87	0.56	0.06
oute	13	5.00	5.00	4.94	4.61	2.70	0.57	0.06
l) Se	15	4.64	4.62	4.55	4.27	2.57	0.55	0.06
	16	2.13	2.13	2.11	2.06	1.55	0.47	0.06
\frown		-						

selectivity (fraction of tuples accessed)

152.9599 153.2835 152.7586 152.8184 152.7484 qs4 qs3 qs2 qs1 qs0 IIIIPI Over incover incoversione

Not Affected

1	47 40	1761	16 61	13 57	1 30	0.57	0.06
3	12.86	13.57	13.22	11.03	4.16	0.56	0.06
5	10.96	10.94	10.55	9.27	3.75	0.57	0.06
7	7.00	6.96	6.80	6.23	3.17	0.56	0.06
9	6.39	6.38	6.25	5.79	3.11	0.54	0.06
11	5.79	5.82	5.75	5.24	2.87	0.56	0.06
13	5.00	5.00	4.94	4.61	2.70	0.57	0.06
15	4.64	4.62	4.55	4.27	2.57	0.55	0.06
16	2.13	2.13	2.11	2.06	1.55	0.47	0.06
	16 15 13 11 9 7 5 3	162.13154.64135.00115.7996.3977.00510.96312.86	162.132.13154.644.62135.005.00115.795.8296.396.3877.006.96510.9610.94312.8613.57	162.132.132.11154.644.624.55135.005.004.94115.795.825.7596.396.386.2577.006.966.80510.9610.9410.55312.8613.5713.22	162.132.132.112.06154.644.624.554.27135.005.004.944.61115.795.825.755.2496.396.386.255.7977.006.966.806.23510.9610.9410.559.27312.8613.5713.2211.03	162.132.132.112.061.55154.644.624.554.272.57135.005.004.944.612.70115.795.825.755.242.8796.396.386.255.793.1177.006.966.806.233.17510.9610.9410.559.273.75312.8613.5713.2211.034.16	162.132.132.112.061.550.47154.644.624.554.272.570.55135.005.004.944.612.700.57115.795.825.755.242.870.5696.396.386.255.793.110.5477.006.966.806.233.170.56510.9610.9410.559.273.750.57312.8613.5713.2211.034.160.56

selectivity (fraction of tuples accessed)



How far are we?

Four Systems



Commercial Row-store (Standard Row)



Trojan Columns in commercial Row-store



Commercial Row-store with vendor support for column technology (DBMS-Y)



Commercial Column-store (DBMS-Z)

- (a) default TPC-H schema
- (b) tuned schema

Q14	81.207649	30.788114	28.0 22.291128 lineitem	5.0
Q12	92.486038	37.331905	33.0 29.785149	4.3
Q 6	77.589034	8.6532381	16.0 25.845965	4.6







What about query optimization?

Rules out query optimization?

Rules out query optimization?



Rules out query optimization?

NO!

QO with aggregate UDFs [SIGMOD'06] Manimal [WebDB'10] HadoopToSQL [EuroSys'10] Black box QO [VLDB'12]



The UDF Business Model

UDFs

Not just for application-specific code

Integrate core database functionality after the fact

Column layouts are just one example!

Meet customer demands quickly

Provide quick feedback before new product release

performance

slow







Summary

performance

slow











