LSST Database server

Osman AÏDEL

Plan

- Infrastructure
- State of the art
- Post ingestion Procedure
- Indexes

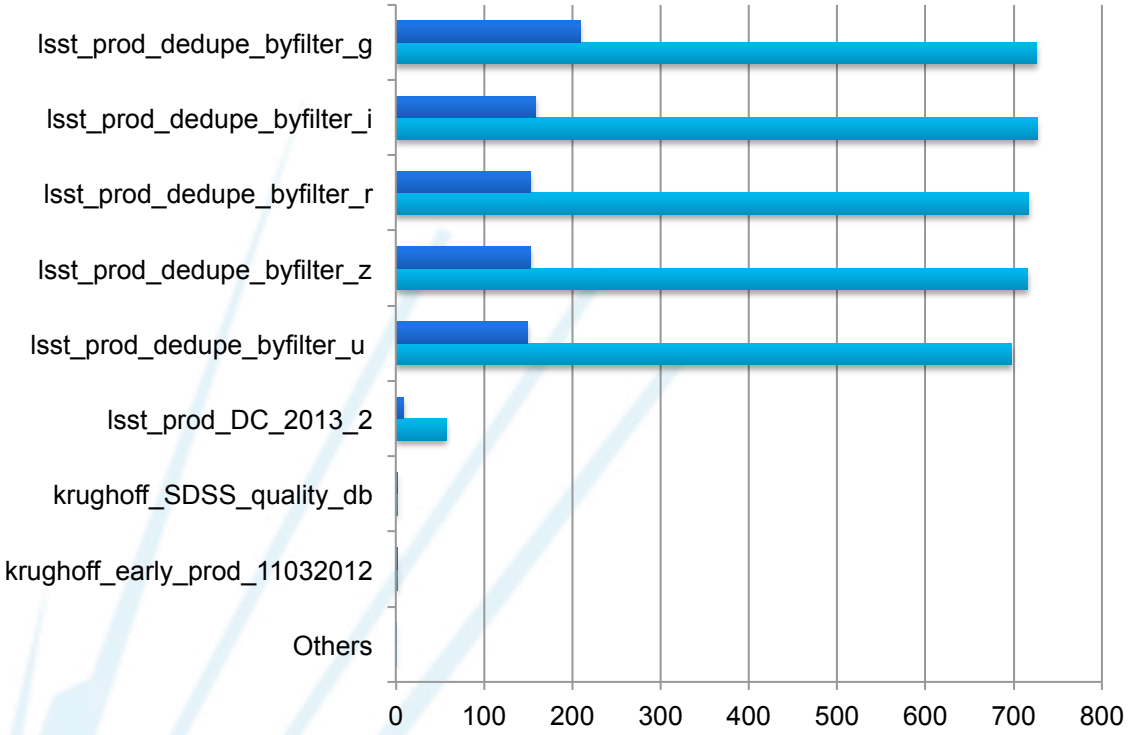
Infrastructure

- Serveur NEC - Express5800/120Rg-1
  - 2 processors Intel(R) Xeon(R) CPU 5160 @ 3.00GHz (Dual core)
  - 40GB RAM
  - 2 SAS disks [15k tpm] of 300 GB (RAID 1)
  - QLogic 4Gb/s FC dual-port card
- Pillar Axiom 600
  - 1 SAN Slammer
  - 8 x Brick SATA v2 (13x2TB)
- File system
  - Ext4 : block size 4KB
  - 12 To allocated in RAID5

Plan

- Infrastructure
- State of the art
- Post ingestion Procedure
- Indexes

State of the art



18 databases
TOP 5 : SDSS DR7
Survey Stripe 82

State of the art

RunDeepForcedSource
- MyISAM Engine - 87 columns - Single Primary key - 7 indexes (keys)
- 2 billion rows - Row size : 395 B - Data size : around 710 GB - Index size : around 150 GB

State of the art

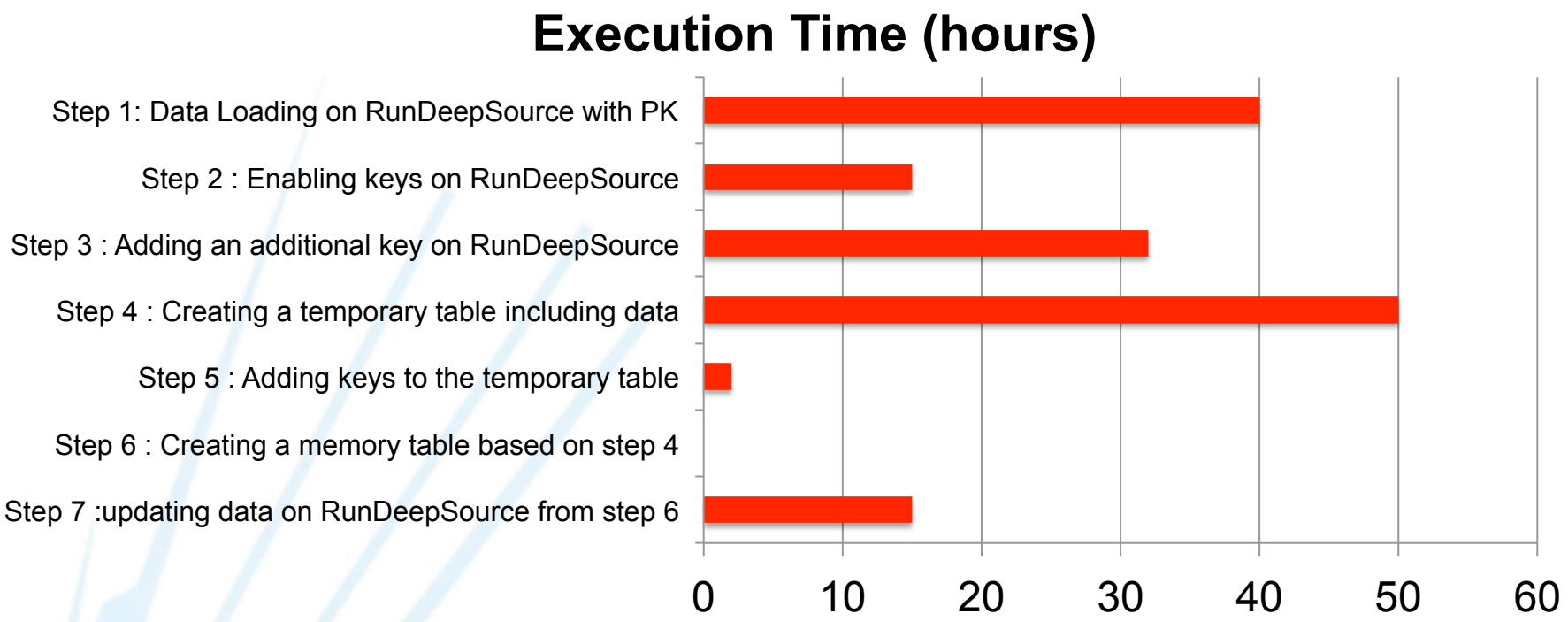
TABLE_SCHEMA	TABLE_TYPE	ENGINE	Nb of objects	Data Size (Gb)	Index Size (Gb)
DB_S13_Stripe82_demo_1	BASE TABLE	MyISAM	38	0.0	0.0
JCT_S13_Stripe82_1	BASE TABLE	MyISAM	38	0.0	0.0
krughoff_early_prod_11032012	BASE TABLE	MyISAM	38	1.7	1.4
krughoff_SDSS_quality_db	BASE TABLE	MyISAM	2	2.0	2.0
lsst	BASE TABLE	InnoDB	36	0.0	0.0
lsst	BASE TABLE	MEMORY	1	0.0	0.0
lsst	BASE TABLE	MyISAM	1	0.0	0.0
lsst_prod_DC_2013_2	BASE TABLE	MyISAM	38	57.3	9.0
lsst_prod_DC_2013_2	VIEW	NULL	1	NULL	NULL
lsst_prod_DC_2013_dirac_test	BASE TABLE	MyISAM	38	0.0	0.0
lsst_prod_DC_2013_one_percent	BASE TABLE	MyISAM	38	0.5	0.0
lsst_prod_DC_2013_one_percent	VIEW	NULL	1	NULL	NULL
lsst_prod_DC_2013_ten_percent	BASE TABLE	MyISAM	38	0.0	0.0
lsst_prod_dedupe_byfilter_g	BASE TABLE	MyISAM	2	726.3	209.3
lsst_prod_dedupe_byfilter_g	VIEW	NULL	1	NULL	NULL
lsst_prod_dedupe_byfilter_i	BASE TABLE	MyISAM	2	727.4	158.5
lsst_prod_dedupe_byfilter_i	VIEW	NULL	1	NULL	NULL
lsst_prod_dedupe_byfilter_r	BASE TABLE	MEMORY	1	0.0	0.0
lsst_prod_dedupe_byfilter_r	BASE TABLE	MyISAM	1	717.5	153.6
lsst_prod_dedupe_byfilter_r	VIEW	NULL	1	NULL	NULL
lsst_prod_dedupe_byfilter_u	BASE TABLE	MyISAM	1	698.2	149.0
lsst_prod_dedupe_byfilter_u	VIEW	NULL	1	NULL	NULL
lsst_prod_dedupe_byfilter_z	BASE TABLE	MyISAM	1	716.2	153.4
lsst_prod_dedupe_byfilter_z	VIEW	NULL	1	NULL	NULL
lsst_prod_dirac_DC2013	BASE TABLE	MyISAM	38	0.0	0.0
lsst_prod_S13_Stripe82_demo_1	BASE TABLE	MyISAM	38	0.0	0.0
scisql_demo	BASE TABLE	MyISAM	4	0.0	0.0
test_fouchez2	BASE TABLE	MyISAM	1	0.0	0.0

Plan

- Infrastructure
- State of the art
- Post ingestion Procedure
- Indexes

Post ingestion procedure

https://dev.lsstcorp.org/trac/wiki/Summer2013/ConfigAndStackTestingPlans/DedupeForcedSources



Total execution time 154 hours (6 days 10 hours)

Post ingestion procedure

- Step 2 : enabling keys on RunDeepForcedSource
  - Mysql needs to sort data before building the index
  - Mysql has two **sort** algorithms for sorting
    - Key cache : recommended for small indexes
    - Filesort : recommended for huge indexes
  - From Mysql client
    - Anyone can run it
    - Very difficult to force the filesort algorithm
    - Ìmpredictible switching from filesort to key cache algorithm

Post ingestion procedure

- Step 2 : enabling keys on RunDeepForcedSource
  - From Myisamchk command
    - Possibility to parallelize the index building
    - More flexible than Mysql client
    - Best performance in mono-thread with myisam_sort_buffer_size=2GB
    - A bug prevents to exceed 4GB on the sort_buffer_size
    - Local access is required
    - Table locked
  - Consuming a lot of disk space (1 TB)
- Step 6 – 7: Updating data on RunDeepForcedSource
  - Both steps were initialy merged into one step
  - Execution time VERY,VERY long > 4 days

Post ingestion procedure

- Procedure only run at CC-IN2P3 :
  - Qserv not yet
  - At NSCA not yet
- Suggestions :
  - Step 1 : Loadind data (40 hours)
    - Create table RunDeepForcedSource without primary key
    - Alter table add primary key
  - Step 2-3 could be grouped together
  - RunDeepForcedSource is, by default, a heap table it might be interesting to sort data at the datafile level
  - Partitionning but on which column ?

Plan

- Infrastructure
- State of the art
- Post ingestion Procedure
- indexes

Indexes

Select * from mytable where id =5;

Without index => full table scan

Important cost

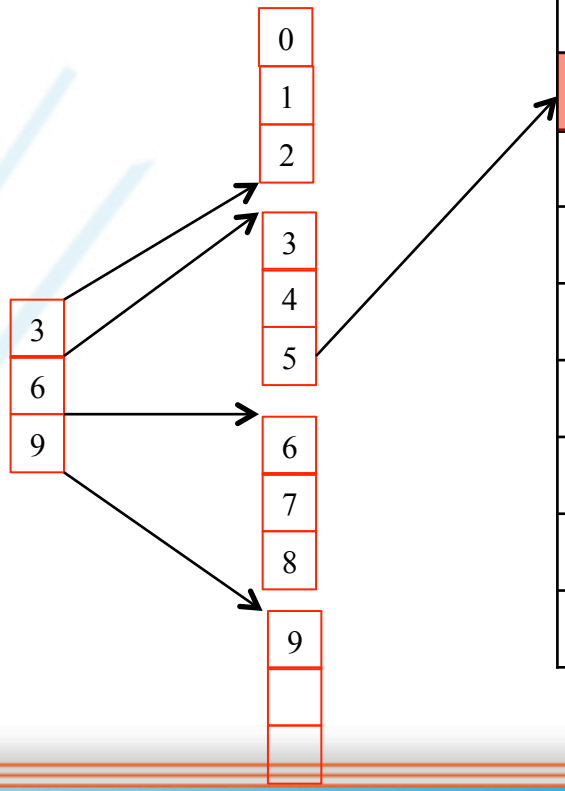
id	First Name	Last name
1	a	a
3	z	z
5	e	e
8	r	r
7	t	t
2	y	y
4	u	u
6	i	i
9	k	k
0	p	p

Indexes

Create index idx on mytable(id);

Select * from mytable where id =5;

Low cost



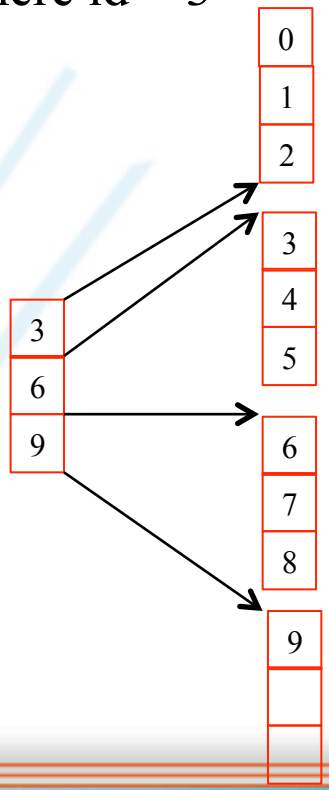
id	First Name	Last name
1	a	a
3	z	z
5	e	e
8	r	r
7	t	t
2	y	y
4	u	u
6	i	i
9	k	k
0	p	p

Indexes

-> Optimizer based on COST
-> Index selectivity = cardinality / Nb of rows
-> example : Index selectivity = 1

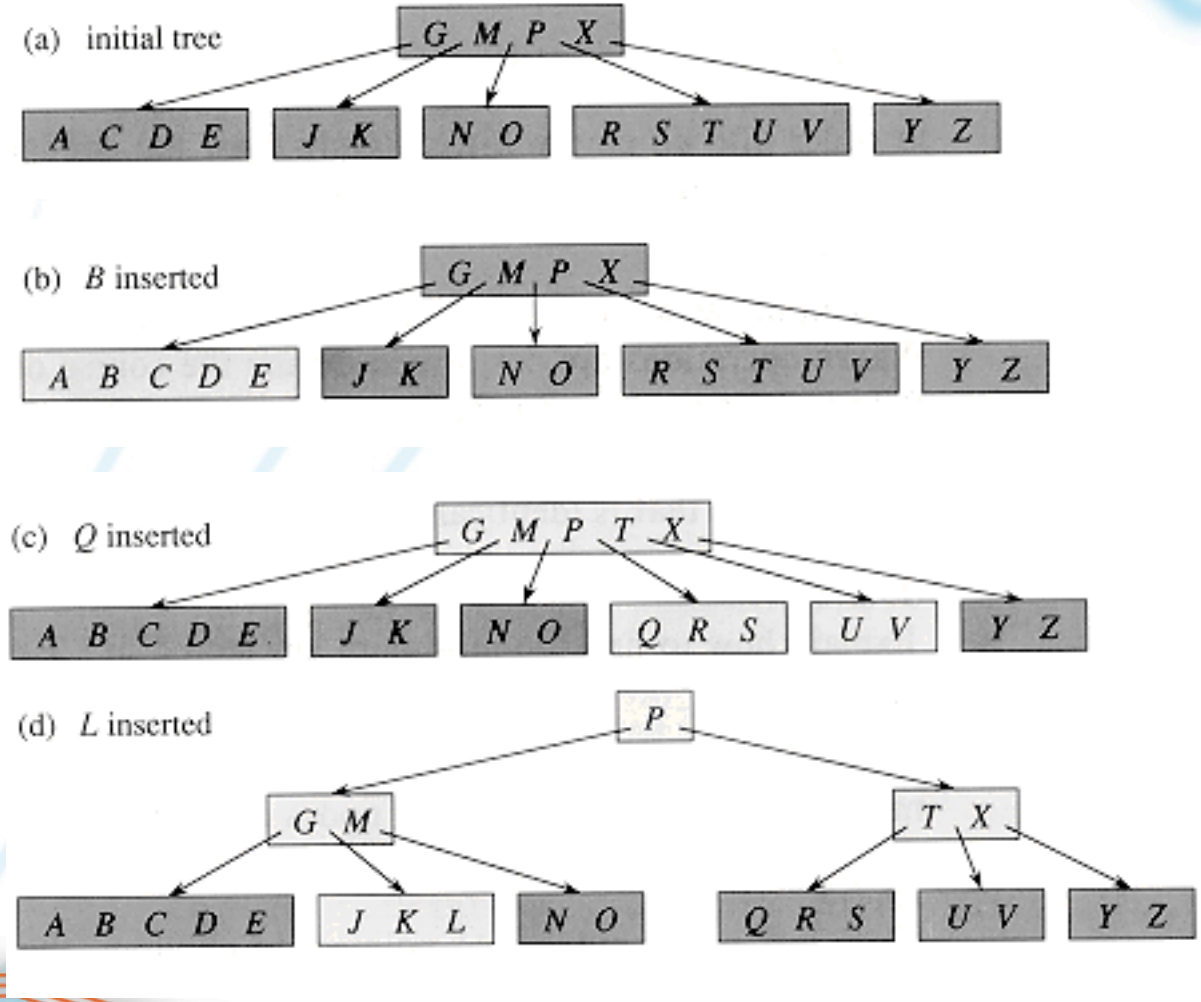
Select * from mytable where id > 3

High Cost -> Full scan

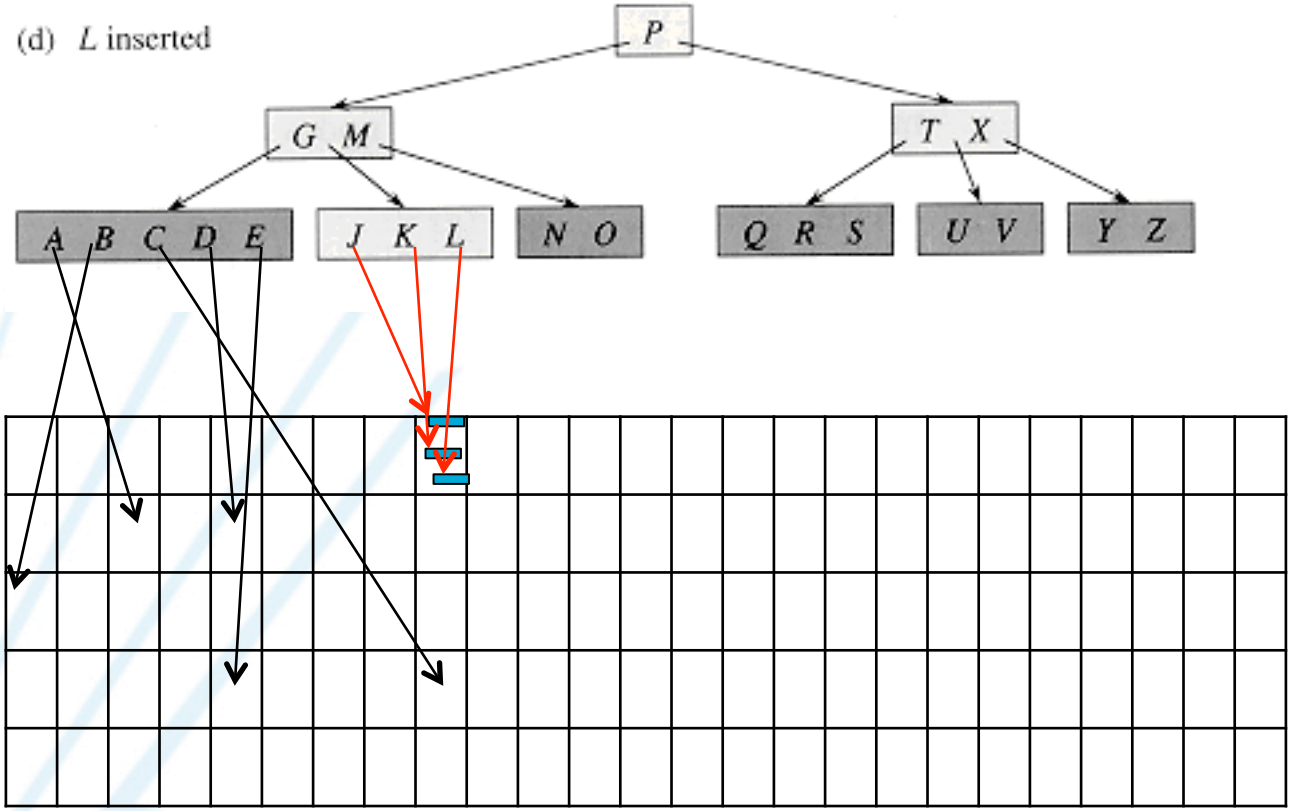


id	First Name	Last name
1	a	a
3	z	z
5	e	e
8	r	r
7	t	t
2	y	y
4	u	u
6	i	i
9	k	k
0	p	p

Indexes

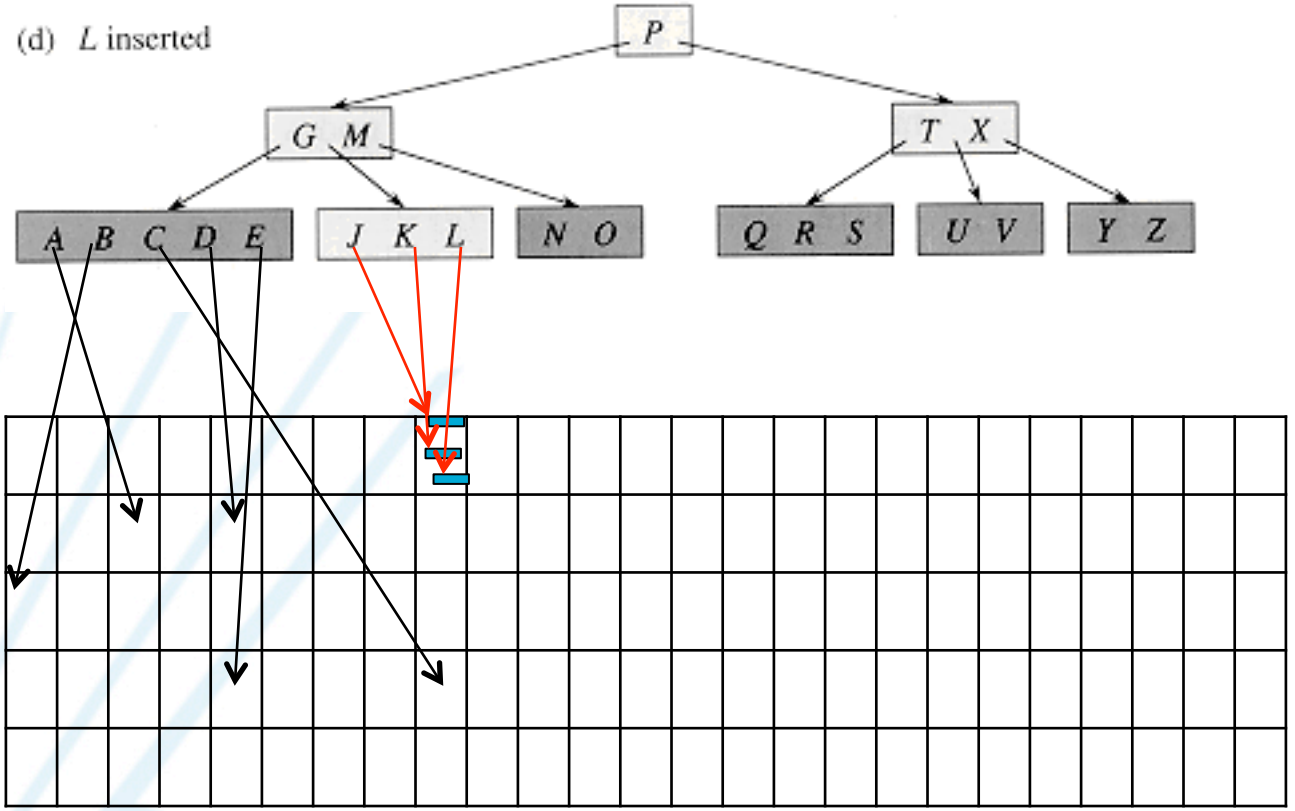

(a) initial tree
G M P X
A C D E
J K
N O
R S T U V
Y Z
(b) B inserted
G M P X
A B C D E
J K
N O
R S T U V
Y Z
(c) Q inserted
G M P T X
A B C D E
J K
N O
Q R S
U V
Y Z
(d) L inserted
P
G M
T X
A B C D E
J K L
N O
Q R S
U V
Y Z

Indexes



Block size : 4Kb

Indexes



Block size : 4Kb