www.questpond.com

The Software Interview Question Bank

-- Maintained by Shivprasad Koirala shiv_koirala@yahoo.com

Do you have a question which can cost somebody a good job mail me at shiv_koirala@yahoo.com

Do you have a suggestion / tips and tricks which can make job searcher easier mail me at shiv_koirala@yahoo.com.

Contents

From the Author

First thing thanks to all those who have sent me complaints and also appreciation for what ever titles i have written till today. But interview question series is very near to my heart as i can understand the pain of searching a job. Thanks to my publishers (BPB) , readers and reviewers to always excuse all my stupid things which i always do.

So why is this PDF free ?. Well i always wanted to distribute things for free specially when its a interview question book which can fetch a job for a developer. But i am also bounded with publishers rules and regulations. And why not they have a whole team of editor, printing guys, designers, distributors, shopkeepers and including me. But again the other aspect, readers should know of what they are buying , the quality and is it really useful to buy this book. So here are sample free questions which i am giving out free to the readers to see the worth of the book.

I can be contacted at shiv_koirala@yahoo.com its bit difficult to answer all answers but as i get time i do it.

We have recently started a career counselling drive absolutely free for new comers and experienced guys. So i have enlisted the following guys on the panel. Thanks to all these guys to accept the panel job of consulting. Feel free to shoot them questions just put a title in the mail saying “Question about Career”. I have always turned up to them when i had some serious career decision to take.

Shivprasad Koirala :- Not a great guy but as i have done the complete book i have to take up one of the positions. You can contact me at shiv_koirala@yahoo.com for technical career aspect.

Tapan Das :- If you think you are aiming at becoming a project manager he is the right person to consult. He can answer all your questions regarding how to groom your career as a project manager tapand@vsnl.com.

Kapil Siddharth :- If you are thinking to grow as architect in a company then he is a guy. When it comes to role model as architect i rate this guy at the top. You can contact him at kapilsiddharth@hotmail.com

Second if you think you can help the developers mail me at shiv_koirala@yahoo.com and if i find you fitting in the panel i will display your mail address. Please note there are no financial rewards as such but i am sure you will be proud of the work you are doing and whos knows what can come up.

Lets make Software Industry a better place to work ..... Happy Job Hunting and Best of Luck

12. Database Optimization

What are indexes?

Index makes your search faster. So defining indexes to your database will make your search faster.

What are B-Trees?

Most of the indexing fundamentals use “B-Tree” or “Balanced-Tree” principle. It’s not a principle that is something is created by SQL Server but is a mathematical derived fundamental.

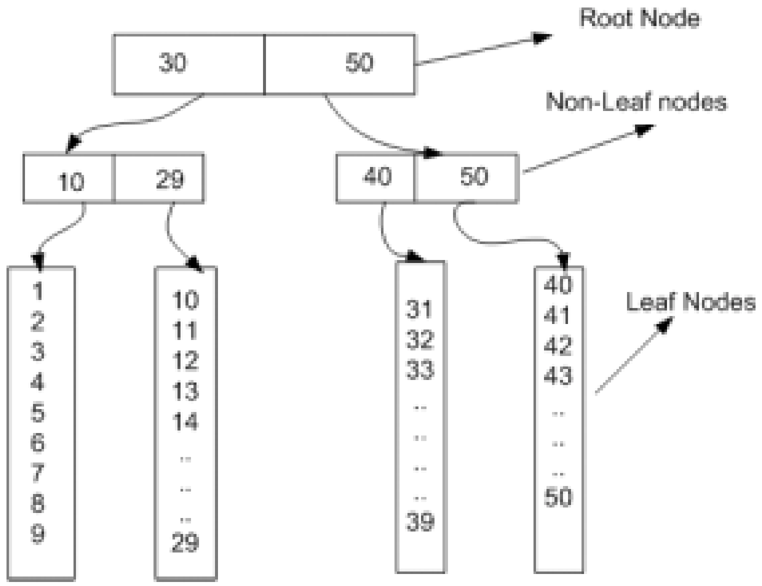


Figure 12.1: - B-Tree principle.

Reading tits and bits is not worth buy complete hard copy see the last page for more details

Above is a sample diagram which explains how B-Tree fundamental works. The above diagram is showing how index will work for number from 1-50. Let's say you want to search 39. SQL Server will first start from the first node i.e. root node.

- √ It will see that the number is greater than 30, so it moves to the 50 node.
- √ Further in Non-Leaf nodes it compares is it more than 40 or less than 40. As it's less than 40 it loops through the leaf nodes which belong to 40 nodes.

You can see that this is all attained in only two steps…faster aaah. That is how exactly indexes work in SQL Server.

I have a table which has lot of inserts, is it a good database design to create indexes on that table?

Twist: - Insert's are slower on tables which have indexes, justify it?

Twist: - Why do page splitting happen?

"B-Tree" stands for balanced tree. In order that "B-tree" fundamental work properly both of the sides should be balanced. All indexing fundamentals in SQL Server use "B-tree" fundamental. Now whenever there is new data inserted or deleted the tree tries to become unbalance. In order that we can understand the fundamental properly let's try to refer the figure down.

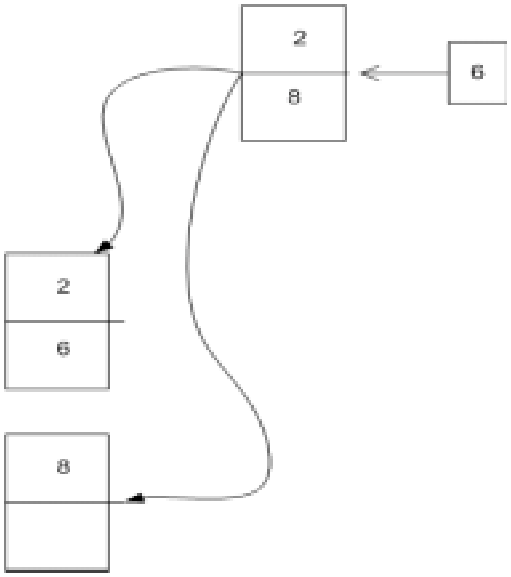


Figure 12.2 : - Page split for Indexed tables

If you see the first level index there is “2” and “8”, now let say we want to insert “6”. In order to balance the “B-TREE” structure rows it will try to split in two pages, as shown. Even though the second page split has some empty area it will go ahead because the primary thing for him is balancing the “B-TREE” for fast retrieval.

Now if you see during the split it is doing some heavy duty here:-

- √ Creates a new page to balance the tree.
- √ Shuffle and move the data to pages.

So if your table is having heavy inserts that means it’s transactional, then you can visualize the amount of splits it will be doing. This will not only increase insert time but will also upset the end-user who is sitting on the screen.

So when you forecast that a table has lot of inserts it’s not a good idea to create indexes.

What are “Table Scan’s” and “Index Scan’s”?

These are ways by which SQL Server searches a record or data in table. In “Table Scan” SQL Server loops through all the records to get to the destination. For instance if you have 1, 2, 5, 23, 63 and 95. If you want to search for 23 it will go through 1, 2 and 5 to reach it. Worst if it wants to search 95 it will loop through all the records.

While for “Index Scan’s” it uses the “B-TREE” fundamental to get to a record. For “B-TREE” refer previous questions.

> *Note: - Which way to search is chosen by SQL Server engine. Example if it finds that the table records are very less it will go for table scan. If it finds the table is huge it will go for index scan.*

What are the two types of indexes and explain them in detail?

> *Twist: - What’s the difference between clustered and non-clustered indexes?*

There are basically two types of indexes:-

- √ Clustered Indexes.
- √ Non-Clustered Indexes.

Ok every thing is same for both the indexes i.e. it uses “B-TREE” for searching data. But the main difference is the way it stores physical data. If you remember the previous figure (give figure number here) there where leaf level and non-leaf level. Leaf level holds the key which is used to identify the record. And non-leaf level actually point to the leaf level.

In clustered index the non-leaf level actually points to the actual data.

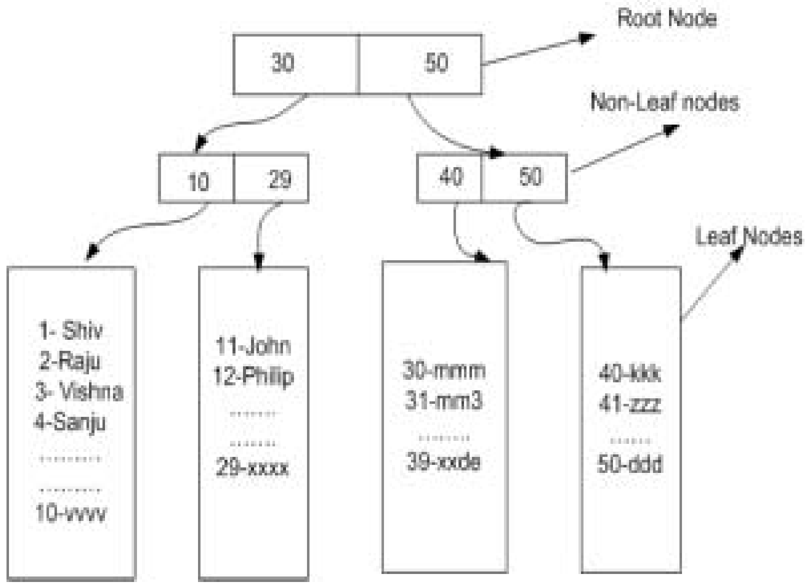


Figure 12.3 : - Clustered Index Architecture

In Non-Clustered index the leaf nodes point to pointers (they are rowid's) which then point to actual data.

Reading tits and bits is not worth buy complete hard copy see the last page for more details

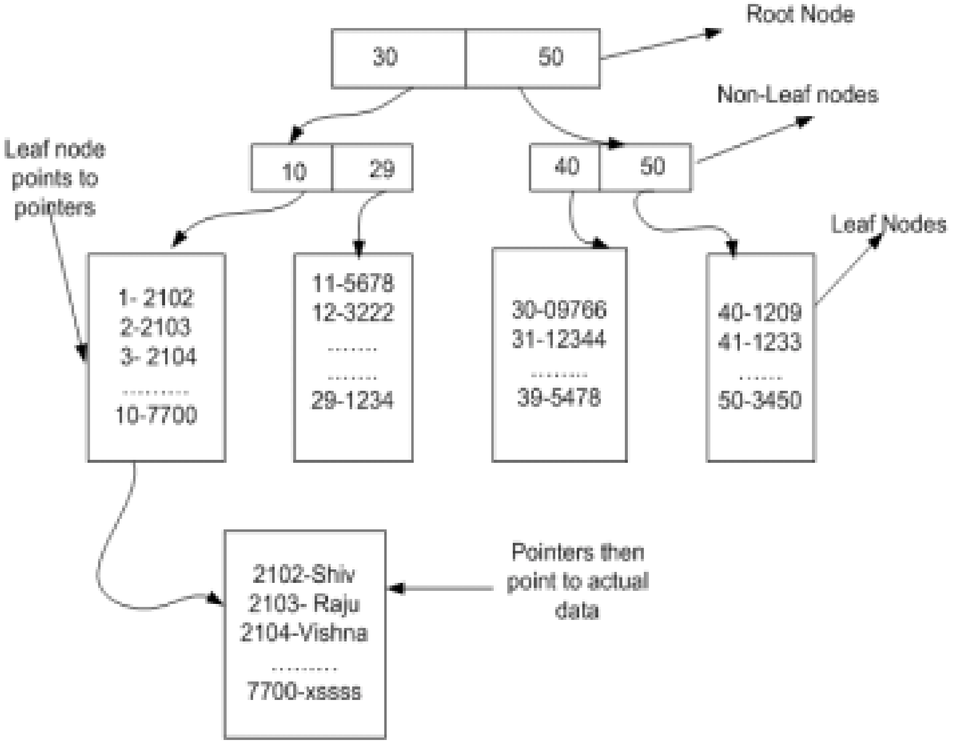


Figure 12.4 : - Non-Clustered Index has pointers.

So here's what the main difference is in clustered and non-clustered , in clustered when we reach the leaf nodes we are on the actual data. In non-clustered indexes we get a pointer, which then points to the actual data.

So after the above fundamentals following are the basic differences between them:-

√ Also note in clustered index actual data as to be sorted in same way as the clustered indexes are. While in non-clustered indexes as we have pointers which is logical arrangement we do need this compulsion.

√ So we can have only one clustered index on a table as we can have only one physical order while we can have more than one non-clustered indexes.

If we make non-clustered index on a table which has clustered indexes, how does the architecture change?

The only change is that the leaf node point to clustered index key. Using this clustered index key can then be used to finally locate the actual data. So the difference is that leaf node has pointers while in the next half it has clustered keys. So if we create non-clustered index on a table which has clustered index it tries to use the clustered index.

(DB)What is "FillFactor" concept in indexes?

When SQL Server creates new indexes, the pages are by default full. "FillFactor" is a percentage value (from 1 – 100) which says how much full your pages will be. By default "FillFactor" value is zero.

(DB) What is the best value for "FillFactor"?

"FillFactor" depends on how transactional your database is. Example if your database is highly transactional (i.e. heavy insert's are happening on the table), then keep the fill factor less around 70. If it's only a read-only database probably used only for reports you specify 100%.

Remember there is a page split when the page is full. So fill factor will play an important role.

What are "Index statistics"?

Statistics are something the query optimizer will use to decide what type of index (table scan or index scan) to be used to search data. Statistics change according to inserts and updates on the table, nature of data on the table etc...In short "Index statistics" are not same in all situations. So DBA has to run statistics again and again after certain interval to ensure that the statistics are up-to-date with the current data.

> *Note: - If you want to create index you can use either the "Create Index" statement or you can use the GUI.*

Reading tits and bits is not worth buy complete hard copy see the last page for more details

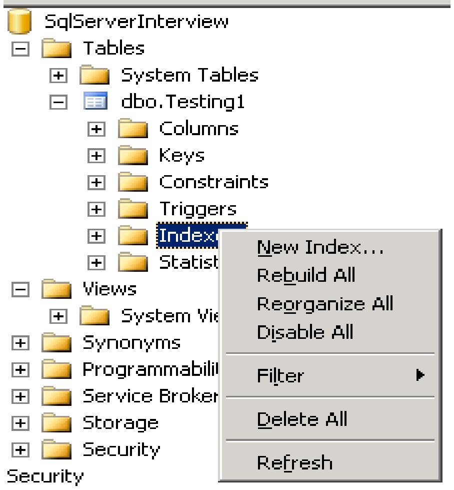


Figure 12.5 : - Index creation in Action.

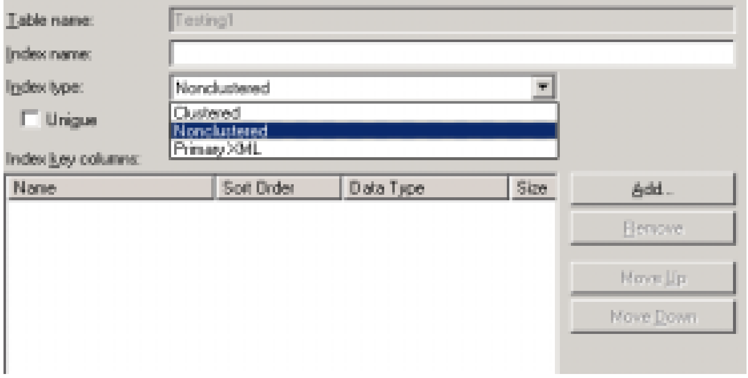


Table name: Testing1

Index name:

Index type: Nonclustered

☐ Unique

Figure 12.6 : - Index Details

(DB)How can we see statistics of an index?

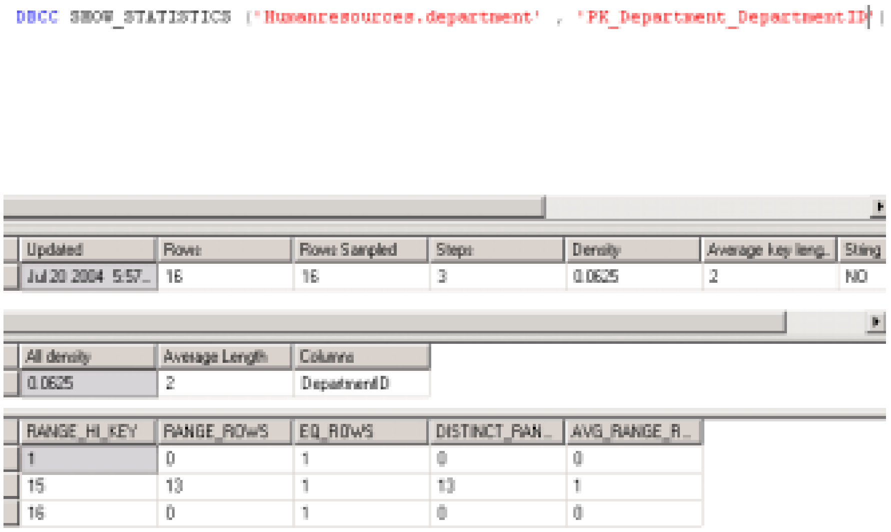
Twist: - How can we measure health of index?

In order to see statistics of any index following the T-SQL command you will need to run.

Note: - Before reading this you should have all the answers of the pervious section clear. Especially about extent, pages and indexes.

```
DECLARE
@ID int,
@IndexID int,
@IndexName varchar(128)
-- input your table and index name
SELECT @IndexName = 'AK_Department_Name'
SET @ID = OBJECT_ID('HumanResources.Department')
SELECT @IndexID = IndID
FROM sysindexes
WHERE id = @ID AND name = @IndexName
--run the DBCC command
DBCC SHOWCONTIG (@id, @IndexID)
```

Just a short note here “DBCC” i.e. “Database consistency checker” is used for checking heath of lot of entities in SQL Server. Now here we will be using it to see index health. After the command is run you will see the following output. You can also run “DBCC SHOWSTATISTICS” to see when was the last time the indexes rebuild.



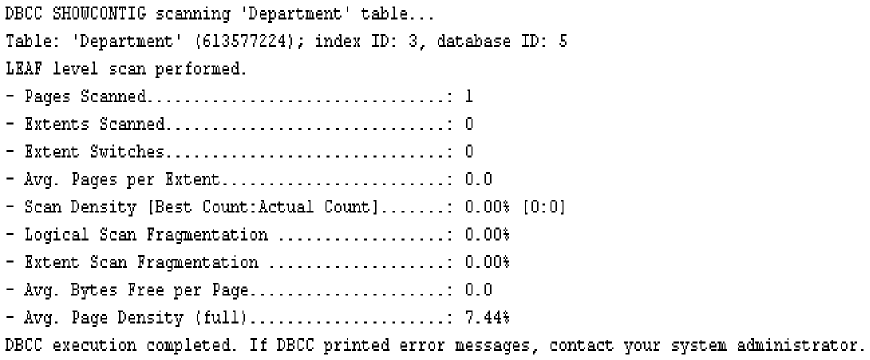
```
DBCC SHOW_STATISTICS ('Humanresources.department' , 'PK_Department_DepartmentID')
```

Updated	Rows	Rows Sampled	Steps	Density	Average key leng...	String
Jul 20 2004 5:57...	16	16	3	0.0625	2	NO

All density	Average Length	Columns
0.0625	2	DepartmentID

RANGE_HI_KEY	RANGE_ROWS	EQ_ROWS	DISTINCT_RAN...	AVG_RANGE_R...
1	0	1	0	0
15	13	1	13	1
16	0	1	0	0

Figure 12.7 DBCC SHOWSTATISTICS



```
DBCC SHOWCONTIG scanning 'Department' table...
Table: 'Department' (613577224); index ID: 3, database ID: 5
LEAF level scan performed.
- Pages Scanned................................: 1
- Extents Scanned..............................: 0
- Extent Switches..............................: 0
- Avg. Pages per Extent........................: 0.0
- Scan Density [Best Count:Actual Count].......: 0.00% [0:0]
- Logical Scan Fragmentation ..................: 0.00%
- Extent Scan Fragmentation ...................: 0.00%
- Avg. Bytes Free per Page.....................: 0.0
- Avg. Page Density (full).....................: 7.44%
DBCC execution completed. If DBCC printed error messages, contact your system administrator.
```

Figure 12.8: - DBCC SHOWCONTIG.

Pages Scanned

The number of pages in the table (for a clustered index) or index.

Extents Scanned

The number of extents in the table or index. If you remember we had said in first instance that extent has pages. More extents for the same number of pages the higher will be the fragmentation.

Extent Switches

The number of times SQL Server moves from one extent to another. More the switches it has to make for the same amount of pages, the more fragmented it is.

Avg. Pages per Extent

The average number of pages per extent. There are eight pages / extent so if you have an extent full with the eight you are in a better position.

Scan Density [Best Count: Actual Count]

This is the percentage ratio of Best count / Actual count. Best count is number of extent changes when everything is perfect. It's like a baseline. Actual count is the actual number of extent changes on that type of scenario.

Logical Scan Fragmentation

Percentage of out-of-order pages returned from scanning the leaf pages of an index. An out of order page is one for which the next page indicated is different page than the page pointed to by the next page pointer in the leaf page. .

Extent Scan Fragmentation

This one is telling us whether an extent is not physically located next to the extent that it is logically located next to. This just means that the leaf pages of your index are not physically in order (though they still can be logically), and just what percentage of the extents this problem pertains to.

Avg. Bytes free per page

This figure tells how many bytes are free per page. If it’s a table with heavy inserts or highly transactional then more free space per page is desirable, so that it will have less page splits.

If it's just a reporting system then having this closer to zero is good as SQL Server can then read data with less number of pages.

Avg. Page density (full)

Average page density (as a percentage). It’s is nothing but:-

1 - (Avg. Bytes free per page / 8096)

8096 = one page is equal to 8096 bytes

> *Note: - Read every of the above sections carefully, incase you are looking for DBA job you will need the above fundamentals to be very clear. Normally interviewer will try to shoot questions like “If you see the fill factor is this much, what will you conclude? , If you see the scan density this much what will you conclude?*

(DB) How do you reorganize your index, once you find the problem?

You can reorganize your index using “DBCC DBREINDEX”. You can either request a particular index to be re-organized or just re-index the all indexes of the table.

This will re-index your all indexes belonging to “HumanResources.Department”.

DBCC DBREINDEX ([HumanResources.Department])

This will re-index only “AK_Department_Name”.

DBCC DBREINDEX ([HumanResources.Department],[AK_Department_Name])

This will re-index with a “fill factor”.

DBCC DBREINDEX ([HumanResources.Department],[AK_Department_Name],70)

You can then again run DBCC SHOWCONTIG to see the results.

What is Fragmentation?

Speed issues occur because of two major things

- √ Fragmentation.
- √ Splits.

Splits have been covered in the first questions. But one other big issue is fragmentation. When database grows it will lead to splits, but what happens when you delete something from the database…HeHeHe life has lot of turns right. Ok let's say you have two extents and each have two pages with some data. Below is a graphical representation. Well actually that's now how things are inside but for sake of clarity lot of things have been removed.

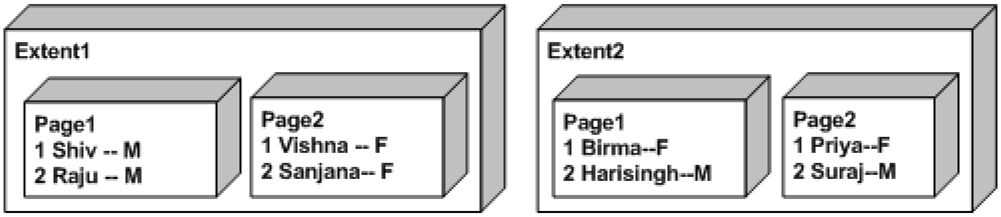


Figure 12.9 : - Data Distribution in Initial Stages

Now over a period of time some Extent and Pages data undergo some delete. Here's the modified database scenario. Now one observation you can see is that some page's are not removed even when they do not have data. Second If SQL server wants to fetch all "Females" it has to span across to two extent and multiple pages within them. This is called as "Fragmentation" i.e. to fetch data you span across lot of pages and extents. This is also termed as "Scattered Data".

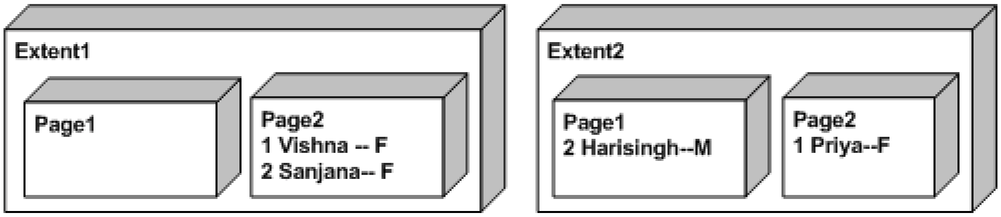


Figure 12.10 : - Data Distribution after Deletes

What if the fragmentation is removed, you only have to search in two extent and two pages. Definitely this will be faster as we are spanning across less entities.

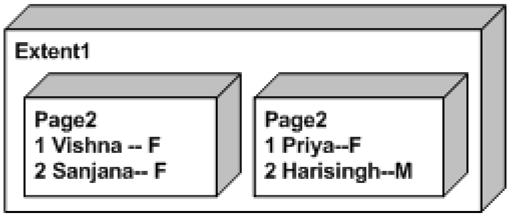


Figure 12.11 : - Fragmentation removed

(DB)How can we measure Fragmentation?

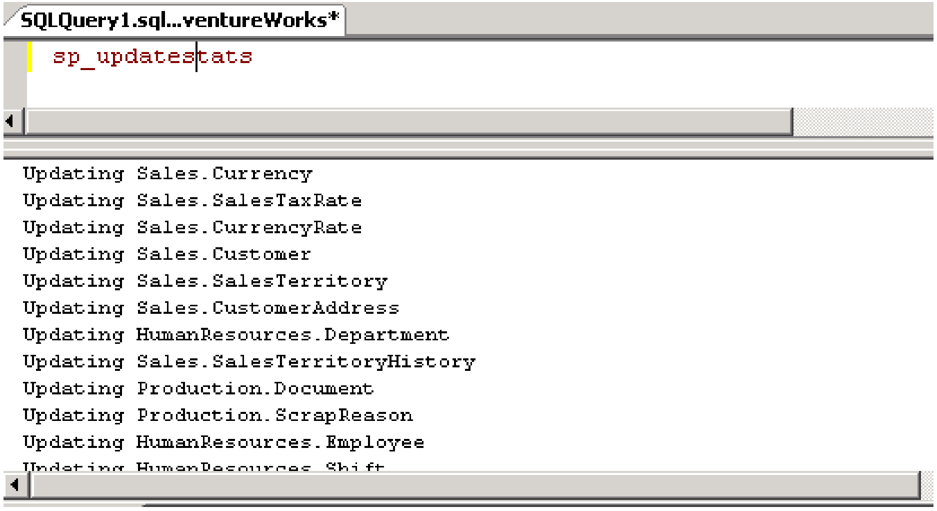
Using "DBCC SHOWCONTIG ".

(DB)How can we remove the Fragmented spaces?

- Update Statistics :- The most used way by DBA's
- Sp_updatestats. :- It's same as update statistics , but update statistics applies only for specified object and indexes , while "sp_updatestats" loops through all tables and applies statistics updates to each and every table. Below is a sample which is run on "AdventureWorks" database.

> *Note: - "AdventureWorks" is a sample database which is shipped with SQL Server 2005.*

Reading tits and bits is not worth buy complete hard copy see the last page for more details



```
sp_updatestats
```

```
Updating Sales.Currency
Updating Sales.SalesTaxRate
Updating Sales.CurrencyRate
Updating Sales.Customer
Updating Sales.SalesTerritory
Updating Sales.CustomerAddress
Updating HumanResources.Department
Updating Sales.SalesTerritoryHistory
Updating Production.Document
Updating Production.ScrapReason
Updating HumanResources.Employee
Updating HumanResources.Shift
```

Figure 12.12: - sp_updatestats in action

√ DBCC INDEXFRAG: - This is not the effective way of doing fragmentation it only does fragmenting on the leaf nodes.

What are the criteria you will look in to while selecting an index?

Note: - Some answers what I have got for this question.

√ I will create index wherever possible.

√ I will create clustered index on every table.

That's why DBA's are always needed.

√ How often the field is used for selection criteria. For example in a "Customer" table you have "CustomerCode" and "PinCode". Most of the searches are going to be performed on "CustomerCode" so it's a good candidate for indexing rather than using "PinCode". In short you can look in to the "WHERE" clauses of SQL to figure out if it's a right choice for indexing.

- √ If the column has higher level of unique values and is used in selection criteria again is a valid member for creating indexes.
- √ If "Foreign" key of table is used extensively in joins (Inner, Outer, and Cross) again a good member for creating indexes.
- √ If you find the table to be highly transactional (huge insert, update and deletes) probably not a good entity for creating indexes. Remember the split problems with Indexes.
- √ You can use the "Index tuning wizard" for index suggestions.

(DB)What is "Index Tuning Wizard"?

Twist: - What is "Work Load File"?

In the previous question the last point was using the "Index Tuning wizard". You can get the "Index Tuning Wizard" from "Microsoft SQL Server Management Studio" – "Tools" – "SQL Profiler".

Note: - This book refers to SQL Server 2005, so probably if you have SQL Server 2000 installed you will get the SQL Profiler in Start – Programs – Microsoft SQL Server -- Profiler. But in this whole book we will refer only SQL Server 2005.We will going step by step for this answer explaining how exactly "Index Tuning Wizard" can be used.

Ok before we define any indexes let's try to understand what is "Work Load File". "Work Load File" is the complete activity that has happened on the server for a specified period of time. All the activity is entered in to a ".trc" file which is called as "Trace File". Later "Index Tuning Wizard" runs on the "Trace File" and on every query fired it tries to find which columns are valid candidates for indexes depending on the Indexes.

Following are the step's to use "Index Tuning Wizard":-

- √ Create the Trace File using "SQL Profiler".
- √ Then use "Database Tuning Advisor" and the "Trace File" for what columns to be indexed.

Create Trace File

Once you have opened the "SQL Profiler" click on "New Trace".

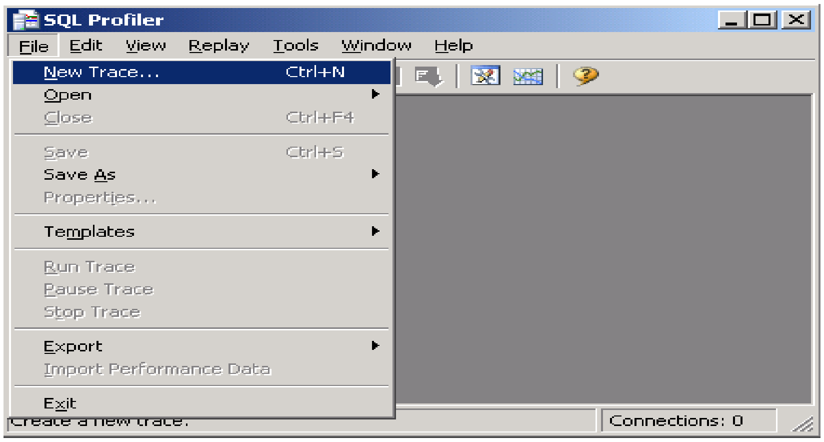
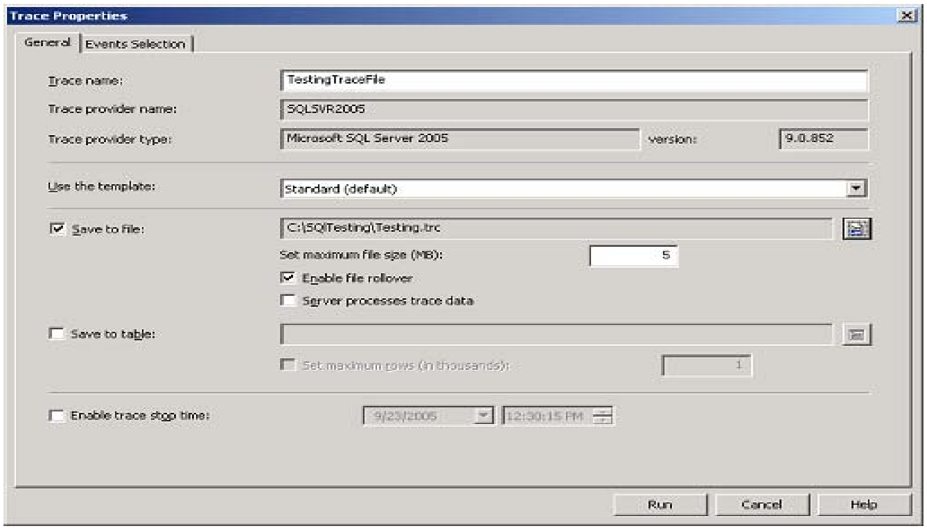


Figure 12.13 : - Create New Trace File.

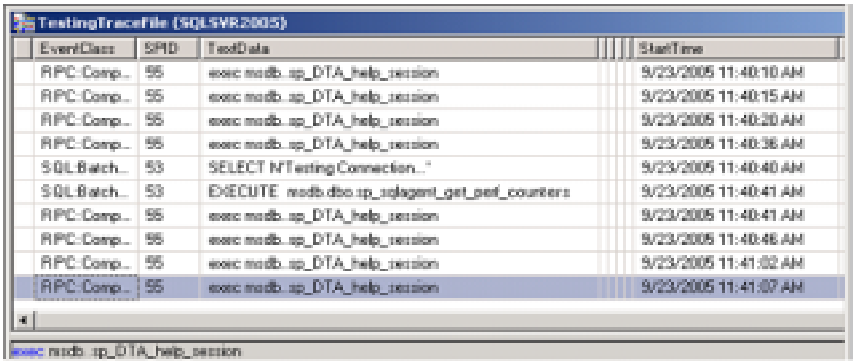
It will alert for giving you all trace file details for instance the "Trace Name", "File where to save". After providing the details click on "Run" button provided below. I have provided the file name of the trace file as "Testing.trc" file.



Trace name: TestingTraceFile
Trace provider name: SQLSVR2005
Trace provider type: Microsoft SQL Server 2005 version: 9.0.852
Use the template: Standard (default)
☑ Save to file: C:\SQlTesting\Testing.trc
Set maximum file size (MB): 5
☑ Enable file rollover
☐ Server processes trace data
☐ Save to table:
☐ Set maximum rows (in thousands): 1
☐ Enable trace stop time: 9/23/2005 12:30:15 PM

Figure 12.14 : - Trace File Details

HUH and the action starts. You will notice that profiler has started tracing queries which are hitting “SQL Server” and logging all those activities in to the “Testing.trc” file. You also see the actual SQL and the time when the SQL was fired.



TestingTraceFile (SQLSVR2005)

EventClass	SPID	TextData	StartTime
RPC:Comp...	55	exec msdb..sp_DTA_help_session	9/23/2005 11:40:10 AM
RPC:Comp...	55	exec msdb..sp_DTA_help_session	9/23/2005 11:40:15 AM
RPC:Comp...	56	exec msdb..sp_DTA_help_session	9/23/2005 11:40:20 AM
RPC:Comp...	56	exec msdb..sp_DTA_help_session	9/23/2005 11:40:36 AM
SQL:Batch...	53	SELECT N'Testing Connection...'	9/23/2005 11:40:40 AM
SQL:Batch...	53	EXECUTE msdb.dbo.sp_sqlagent_get_perf_counters	9/23/2005 11:40:41 AM
RPC:Comp...	55	exec msdb..sp_DTA_help_session	9/23/2005 11:40:41 AM
RPC:Comp...	56	exec msdb..sp_DTA_help_session	9/23/2005 11:40:46 AM
RPC:Comp...	55	exec msdb..sp_DTA_help_session	9/23/2005 11:41:02 AM
RPC:Comp...	55	exec msdb..sp_DTA_help_session	9/23/2005 11:41:07 AM

```
exec msdb..sp_DTA_help_session
```

Figure 12.15 : - Tracing in Actions

Let the trace run for some but of time. In actually practical environment I run the trace for almost two hours in peak to capture the actual load on server. You can stop the trace by clicking on the red icon given above.

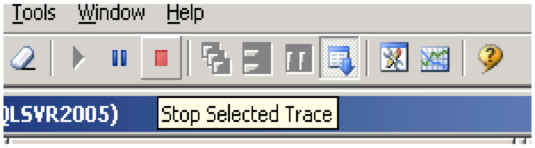


Figure 12.16 : - Stop Trace File.

You can go the folder and see your “.trc” file created. If you try to open it in notepad you will see binary data. It can only be opened using the profiler. So now that we have the load file we have to just say to the advisor hey advisor here’s my problem (trace file) can you suggest me some good indexes to improve my database performance.

Using Database Tuning Advisor

In order to go to “Database Tuning Advisor” you can go from “Tools” – “Database Tuning Advisor”.

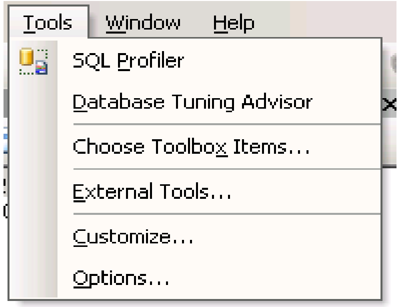


Figure 12.17 : - Menu for SQL Profiler and Database advisor

In order to supply the work load file you have to start a new session in “Database tuning advisor”.

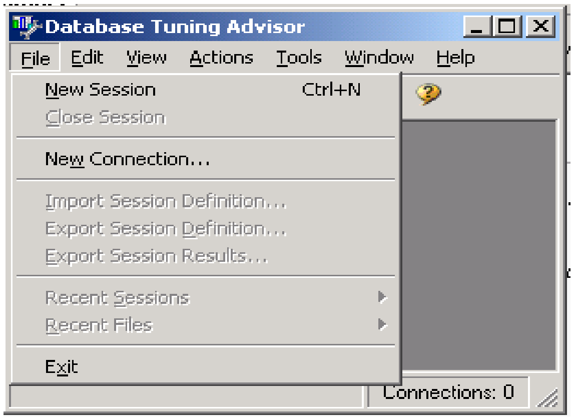


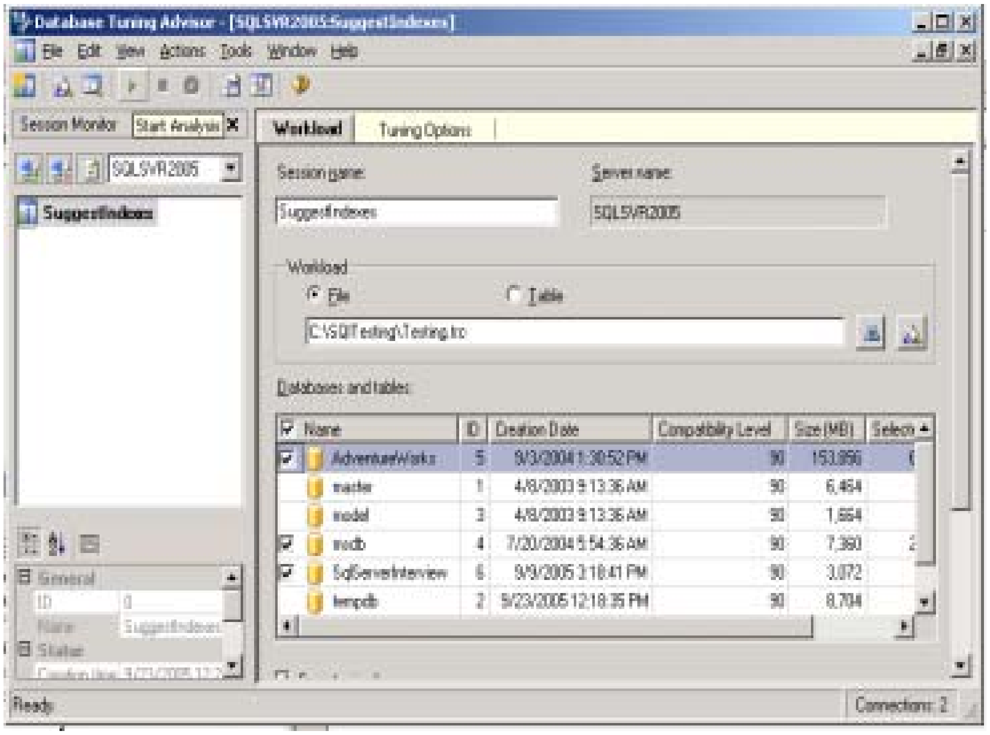
Figure 12.18 : - Creating New Session in Advisor

After you have said “New Session” you have to supply all details for the session. There are two primary requirements you need to provide to the Session:-

- √ Session Name

- √ "Work Load File" or "Table" (Note you can create either a trace file or you can put it in SQL Server table while running the profiler).

I have provided my "Testing.trc" file which was created when I ran the SQL profiler. You can also filter for which database you need index suggestions. At this moment I have checked all the databases. After all the details are filled in you have to click on "Green" icon with the arrow. You can see the tool tip as "Start analysis" in the image below.



Session name: SuggestIndexes

Server name: SQLSVR2005

Workload: ● File ○ Table

C:\SQlTesting\Testing.trc

Databases and tables:

☑	Name	ID	Creation Date	Compatibility Level	Size (MB)
☑	AdventureWorks	5	9/3/2004 1:30:52 PM	90	153,856
	master	1	4/8/2003 9:13:36 AM	90	6,464
	model	3	4/8/2003 9:13:36 AM	90	1,664
☑	msdb	4	7/20/2004 5:54:36 AM	90	7,360
☑	SqlServerInterview	6	9/9/2005 3:18:41 PM	90	3,072
	tempdb	2	9/23/2005 12:18:35 PM	90	8,704

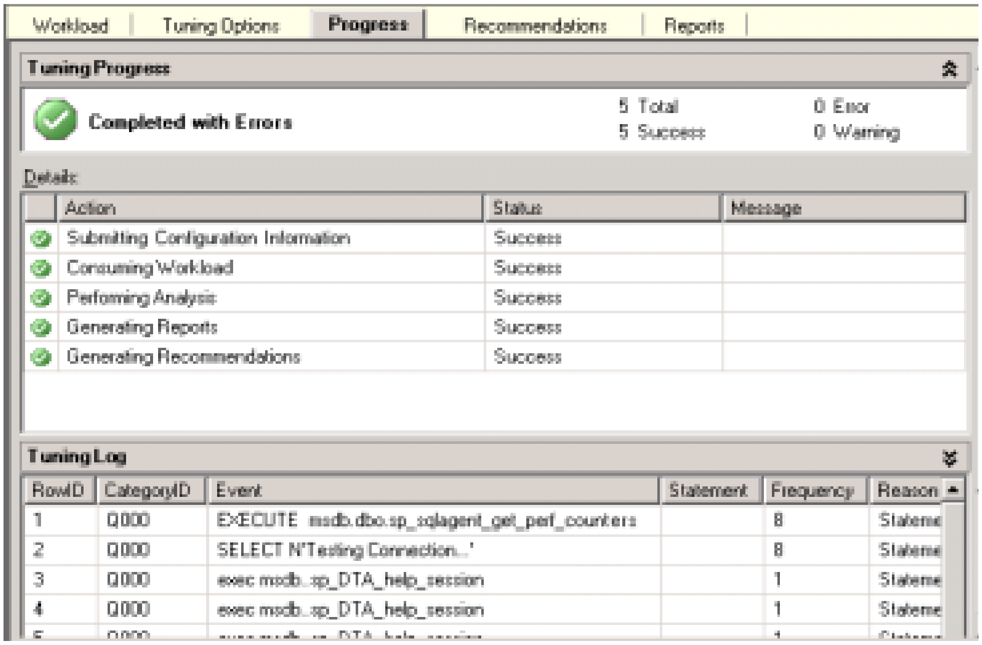
Figure 12.19 : - Session Details for Advisor

While analyzing the trace file it performs basic four major steps:-

- √ Submits the configuration information.
- √ Consumes the Work load data (that can be in format of a file or a database table).

- √ Start performing analysis on all the SQL executed in the trace file.
- √ Generates reports based on analysis.
- √ Finally give the index recommendations.

You can see all the above steps have run successfully which is indicated by “0 Error and 0 Warning”.



Tuning Progress

Completed with Errors

5 Total
5 Success
0 Error
0 Warning

Details:

	Action	Status	Message
✓	Submiting Configuration Information	Success	
✓	Consuming Workload	Success	
✓	Perfoming Analysis	Success	
✓	Generating Reports	Success	
✓	Generating Recommendations	Success	

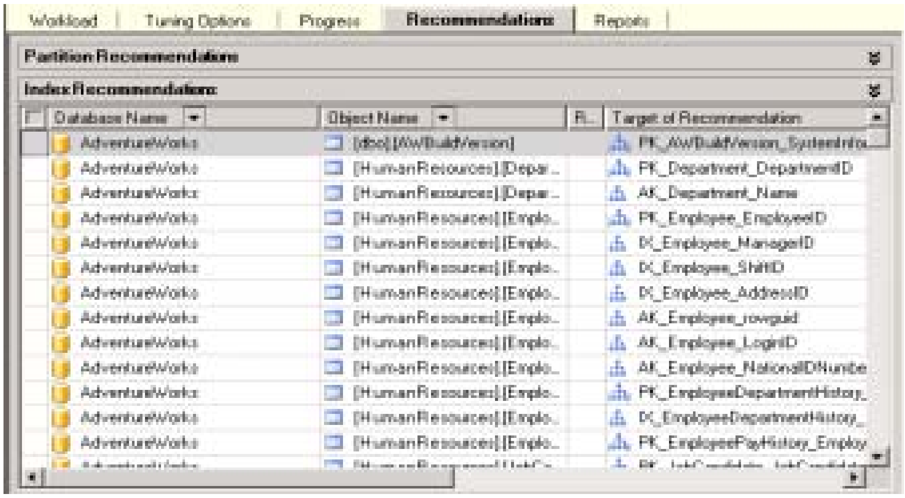
Tuning Log

RowID	CategoryID	Event	Statement	Frequency	Reason
1	Q000	EXECUTE msdb.dbo.sp_sqlagent_get_perf_counters		8	Stateme
2	Q000	SELECT N'Testing Connection...'		8	Stateme
3	Q000	exec msdb..sp_DTA_help_session		1	Stateme
4	Q000	exec msdb..sp_DTA_help_session		1	Stateme

Figure 12.20 : - Session completed with out Errors

Now its time to see what index recommendations SQL Server has provided us. Also note it has included two new tabs after the analysis was done “Recommendations” and “Reports”.

You can see on “AdventureWorks” SQL Server has given me huge recommendations. Example on “HumanResources.Department” he has told me to create index on “PK_Department_DepartmentId”.

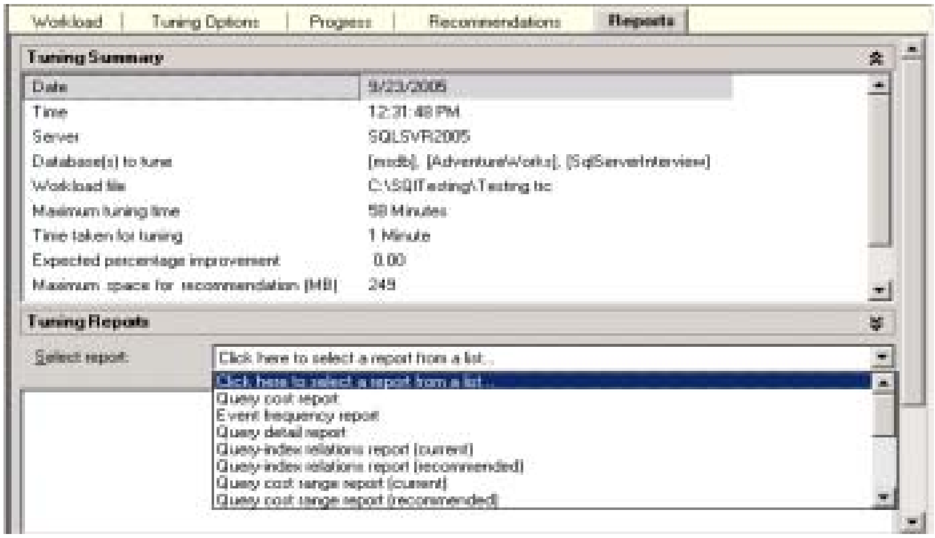


☐	Database Name	Object Name	R...	Target of Recommendation
	AdventureWorks	[dbo].[AWBuildVersion]		PK_AWBuildVersion_SystemInfo...
	AdventureWorks	[HumanResources].[Depar...		PK_Department_DepartmentID
	AdventureWorks	[HumanResources].[Depar...		AK_Department_Name
	AdventureWorks	[HumanResources].[Emplo...		PK_Employee_EmployeeID
	AdventureWorks	[HumanResources].[Emplo...		IX_Employee_ManagerID
	AdventureWorks	[HumanResources].[Emplo...		IX_Employee_ShiftID
	AdventureWorks	[HumanResources].[Emplo...		IX_Employee_AddressID
	AdventureWorks	[HumanResources].[Emplo...		AK_Employee_rowguid
	AdventureWorks	[HumanResources].[Emplo...		AK_Employee_LoginID
	AdventureWorks	[HumanResources].[Emplo...		AK_Employee_NationalIDNumbe
	AdventureWorks	[HumanResources].[Emplo...		PK_EmployeeDepartmentHistory_
	AdventureWorks	[HumanResources].[Emplo...		IX_EmployeeDepartmentHistory_
	AdventureWorks	[HumanResources].[Emplo...		PK_EmployeePayHistory_Employ

Figure 12.21 : - Recommendations by SQL Server

In case you want to see detail reports you can click on the “Reports” tab and there are wide range of reports which you can use to analyze how you database is performing on that “Work Load” file.

Reading tits and bits is not worth buy complete hard copy see the last page for more details


Workload
Tuning Options
Progress
Recommendations
Reports
Tuning Summary
Tuning Reports
Select report:
Click here to select a report from a list...
Click here to select a report from a list...
Query cost report
Event frequency report
Query detail report
Query-index relations report (current)
Query-index relations report (recommended)
Query cost range report (current)
Query cost range report (recommended)

Date	9/23/2005
Time	12:31:48 PM
Server	SQLSVR2005
Database(s) to tune	[msdb], [AdventureWorks], [SqlServerInterview]
Workload file	C:\SQlTesting\Testing.trc
Maximum tuning time	58 Minutes
Time taken for tuning	1 Minute
Expected percentage improvement	0.00
Maximum space for recommendation (MB)	249

Figure 12.22 : - Reports by Advisor

Note: - The whole point of putting this all step by step was that you have complete understanding of how to do "automatic index decision" using SQL Server. During interview one of the question's that is very sure "How do you increase speed performance of SQL Server? " and talking about the "Index Tuning Wizard" can fetch you some decent points.

(DB)What is an Execution plan?

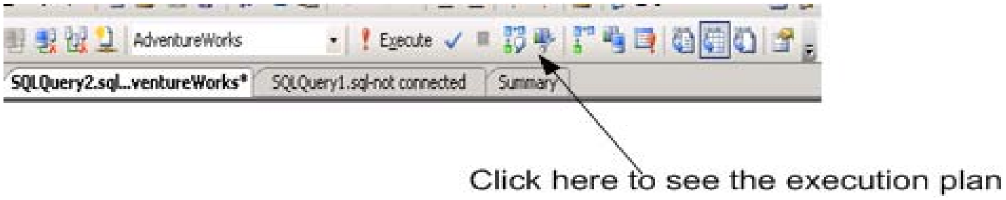
Execution plan gives how the query optimizer will execute a give SQL query. Basically it shows the complete plan of how SQL will be executed. The whole query plan is prepared depending on lot of data for instance:-

√ What type of indexes do the tables in the SQL have?

√ Amount of data.

√ Type of joins in SQL (Inner join , Left join , Cross join , Right join etc)

Click on the ICON in SQL Server management studio as shown in figure below.

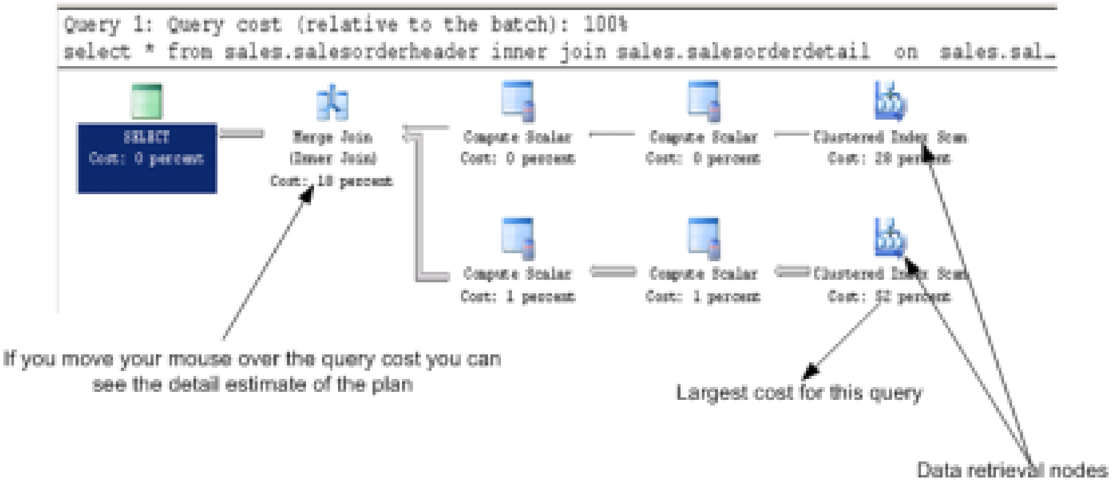


Click here to see the execution plan

Figure 12.23 : - Click here to see execution plan

In bottom window pane you will see the complete break up of how your SQL Query will execute. Following is the way to read it:-

- √ Data flows from left to right.
- √ Any execution plan sums to total 100 %. For instance in the below figure it is 18 + 28 + 1 + 1 + 52. So the highest is taken by Index scan 52 percent. Probably we can look in to that logic and optimize this query.
- √ Right most nodes are actually data retrieval nodes. I have shown them with arrows the two nodes.
- √ In below figure you can see some arrows are thick and some are thin. More the thickness more the data is transferred.
- √ There are three types of join logic nested join, hash join and merge join.



```
Query 1: Query cost (relative to the batch): 100%
select * from sales.salesorderheader inner join sales.salesorderdetail  on  sales.sal…
```

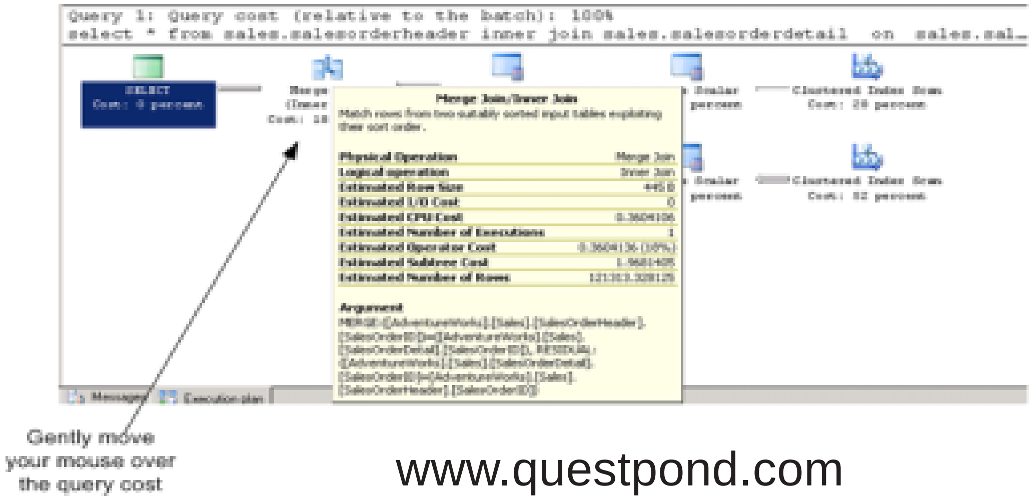
If you move your mouse over the query cost you can see the detail estimate of the plan

Largest cost for this query

Data retrieval nodes

Figure 12.24 : - Largest Cost Query

If you move your mouse gently over any execution strategy you will see a detail breakup of how that node is distributed.



```
Query 1: Query cost (relative to the batch): 100%
select * from sales.salesorderheader inner join sales.salesorderdetail  on  sales.sal…
```

Merge Join/Inner Join

Match rows from two suitably sorted input tables exploiting their sort order.

Physical Operation	Merge Join
Logical operation	Inner Join
Estimated Row Size	445 B
Estimated I/O Cost	0
Estimated CPU Cost	0.3604106
Estimated Number of Executions	1
Estimated Operator Cost	0.3604136 (18%)
Estimated Subtree Cost	1.9681405
Estimated Number of Rows	121313.328125

Argument

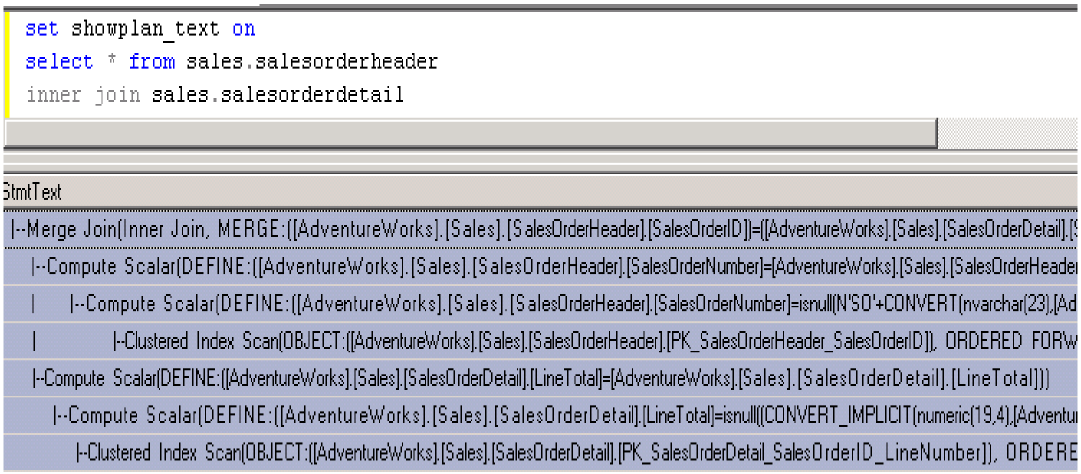
MERGE:([AdventureWorks].[Sales].[SalesOrderHeader].[SalesOrderID])=([AdventureWorks].[Sales].[SalesOrderDetail].[SalesOrderID]), RESIDUAL:([AdventureWorks].[Sales].[SalesOrderDetail].[SalesOrderID]=[AdventureWorks].[Sales].[SalesOrderHeader].[SalesOrderID])

Gently move your mouse over the query cost

Figure 12.25 : - Complete Break up estimate

How do you see the SQL plan in textual format?

Execute the following "set showplan_text on" and after that execute your SQL, you will see a textual plan of the query. In the first question what I discussed was a graphical view of the query plan. Below is a view of how a textual query plan looks like. In older versions of SQL Server where there was no way of seeing the query plan graphically "SHOWPLAN" was the most used. Today if any one is using it that I think he is doing a show business or a new come learner.



```
set showplan_text on
select * from sales.salesorderheader
inner join sales.salesorderdetail
```

```
StmtText
|--Merge Join(Inner Join, MERGE:([AdventureWorks].[Sales].[SalesOrderHeader].[SalesOrderID])=([AdventureWorks].[Sales].[SalesOrderDetail].[
  |--Compute Scalar(DEFINE:([AdventureWorks].[Sales].[SalesOrderHeader].[SalesOrderNumber]=[AdventureWorks].[Sales].[SalesOrderHeade
  |    |--Compute Scalar(DEFINE:([AdventureWorks].[Sales].[SalesOrderHeader].[SalesOrderNumber]=isnull(N'SO'+CONVERT(nvarchar(23),[Ad
  |         |--Clustered Index Scan(OBJECT:([AdventureWorks].[Sales].[SalesOrderHeader].[PK_SalesOrderHeader_SalesOrderID]), ORDERED FORW
  |--Compute Scalar(DEFINE:([AdventureWorks].[Sales].[SalesOrderDetail].[LineTotal]=[AdventureWorks].[Sales].[SalesOrderDetail].[LineTotal]))
     |--Compute Scalar(DEFINE:([AdventureWorks].[Sales].[SalesOrderDetail].[LineTotal]=isnull((CONVERT_IMPLICIT(numeric(19,4),[Adventu
        |--Clustered Index Scan(OBJECT:([AdventureWorks].[Sales].[SalesOrderDetail].[PK_SalesOrderDetail_SalesOrderID_LineNumber]), ORDERE
```

Figure 12.26 : - Textual Query Plan View

(DB)What is nested join, hash join and merge join in SQL Query plan?

A join is whenever two inputs are compared to determine and output. There are three basic types of strategies for this and they are: nested loops join, merge join and hash join. When a join happens the optimizer determines which of these three algorithms is best to use for the given problem, however any of the three could be used for any join. All of the costs related to the join are analyzed the most cost efficient algorithm is picked for use. These are in-memory loops used by SQL Server.

Nested Join

If you have less data this is the best logic. It has two loops one is the outer and the other is the inner loop. For every outer loop, its loops through all records in the inner loop. You can see the two loop inputs given to the logic. The top index scan is the outer loop and bottom index seek is the inner loop for every outer record.

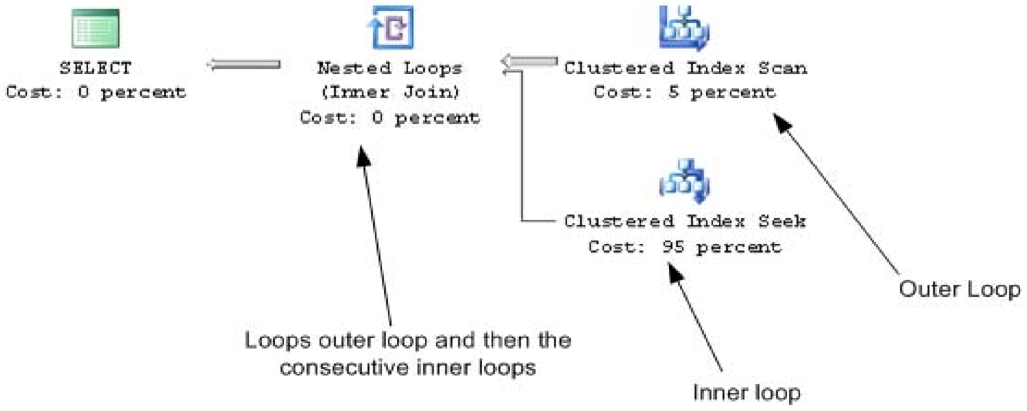


Figure 12.27 : - Nested joins

It's like executing the below logic:-

```
For each outer records
    For each inner records
    Next
Next
```

So you visualize that if there fewer inner records this is a good solution.

Hash Join

Hash join has two input "Probe" and "Build" input. First the "Build" input is processed and then the "Probe" input. Which ever input is smaller is the "Build" input. SQL Server first builds a hash table using the build table input. After that he loops through the probe input and finds the matches using the hash table created previously using the build table and does the processing and gives the output.

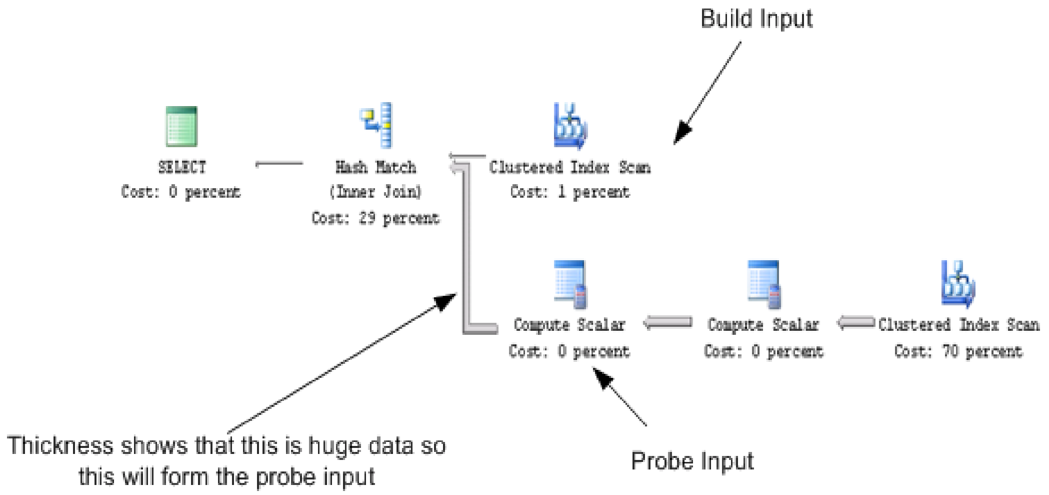


Figure 12.28 : - Hash Join

Merge Join

In merge joins both the inputs are sorted on the merge columns. Merge columns are determined depending on the inner join defined in SQL. Since each input join is sorted merge join takes input and compares for equality. If there is equality then matching row is produced. This is processed till the end of rows.

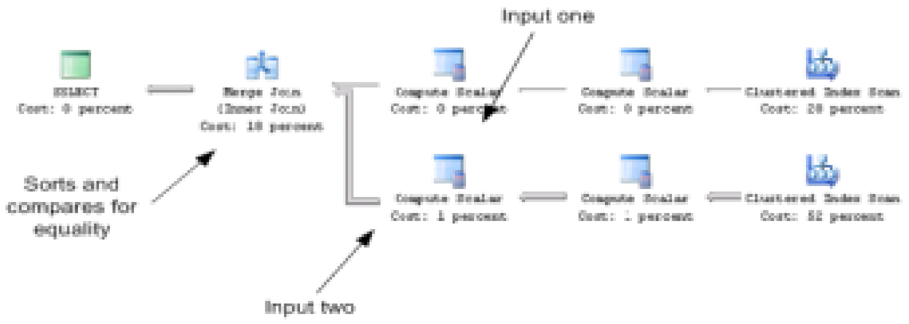


Figure 12.29: - Merge Join

What joins are good in what situations?

Nested joins best suited if the table is small and it’s a must the inner table should have an index.

Merge joins best of large tables and both tables participating in the joins should have indexes.

Hash joins best for small outer tables and large inner tables. Not necessary that tables should have indexes, but would be better if outer table has indexes.

> *Note: - Previously we have discussed about table scan and index scan do revise it which is also important from the aspect of reading query plan.*

(DB)What is RAID and how does it work ?

Redundant Array of Independent Disks (RAID) is a term used to describe the technique of improving data availability through the use of arrays of disks and various data-striping methodologies. Disk arrays are groups of disk drives that work together to achieve higher data-transfer and I/O rates than those provided by single large drives. An array is a set of multiple disk drives plus a specialized controller (an array controller) that keeps track of how data is distributed across the drives. Data for a particular file is written in segments to the different drives in the array rather than being written to a single drive.

For speed and reliability, it's better to have more disks. When these disks are arranged in certain patterns and use a specific controller, they are called a Redundant Array of Inexpensive Disks (RAID) set. There are several numbers associated with RAID, but the most common are 1, 5 and 10.

RAID 1 works by duplicating the same writes on two hard drives. Let's assume you have two 20 Gigabyte drives. In RAID 1, data is written at the same time to both drives. RAID1 is optimized for fast writes.

RAID 5 works by writing parts of data across all drives in the set (it requires at least three drives). If a drive failed, the entire set would be worthless. To combat this problem, one of the drives stores a "parity" bit. Think of a math problem, such as 3 + 7 = 10. You can think of the drives as storing one of the numbers, and the 10 is the parity part. By removing any one of the numbers, you can get it back by referring to the other two, like this: 3 + X = 10. Of course, losing more than one could be evil. RAID 5 is optimized for reads.

RAID 10 is a bit of a combination of both types. It doesn't store a parity bit, so it's fast, but it duplicates the data on two drives to be safe. You need at least four drives for RAID 10. This type of RAID is probably the best compromise for a database server.

Note :- It's difficult to cover complete aspect of RAID in this book.It's better to take some decent SQL SERVER book for in detail knowledge , but yes from interview aspect you can probably escape with this answer.

Reading tits and bits is not worth buy complete hard copy see the last page for more details

Career Path Institute

Author runs the “Softwar Career Path Insitute” personally in mumbai. If you are interested you can contact him regarding admissions at shiv_koirala@yahoo.com. Our courses are mainly targetting from how to get a job perspective.

Below are some of the courses offered :-

-- Interview preparation course two days (Saturday and Sunday Batch). (C# , SQL Server)
-- Full one year course for C# , SQL Server

Distribution Partner

Do you have a news group or website where you want to distribute this PDF free. Contact at shiv_koirala@yahoo.com we will put your logo and send you a complete zipped file which you can host at your site to increase user visits. Be our partner and increase your user visits in your website. We do not take any charge from our partner to distrubute these sample copies. But yes the contents of this PDF can not be modified. If you are our partner you get regular updates of our interview question releases.

Illegal distributions of this PDF will be taken seriously. Just send a mail and be our distribution partner with your website logo proudly do not distribute illegally.

www.questpond.com

How to buy the book

BPB has done a great job of making this book reach to places where i can hardly imagine. But just incase its not near to your place mail bpb@bol.net.in.
If you are from India you can contact one of the shops below:-
MUMBAI-22078296/97/022-22070989
KOLKATA-22826518/19
HYDERABAD-24756967,24756400
BANGALORE-25587923,25584641
AHMEDABAD-26421611
BHATINA(PUNJAB)-2237387,
CHENNAI-28410796,28550491
DELHI/NEW DELHI-23254990/91,23325760,26415092,24691288

Pakistan
M/s. Vanguard Books P Ltd, 45 The Mall, Lahore, Pakistan (Tel: 0092-42-7235767, 7243783 and 7243779 and Fax: 7245097)
E-mail: vbl@brain.net.pk

If you are not from india or pakistan :-
Ray McLennan, director,Motilal (UK) Books of India,367 High Street.
London Colney,
St.Albans, Hertfordshire,AL2 1EA, U.K.
Tel. +44 (0)1727 761 677,Fax.+44 (0)1727 761 357,info@mlbduk.com,www.mlbduk.com

Want to buy from online :-
.NET Interview Questions From Prakash books

http://www.prakashbooks.com/details.php3?id=17875&c=Computer Books

If you want to buy from Amazon

http://www.amazon.co.uk/exec/obidos/ASIN/8183330770/qid%3D1139594062/026-8105897-7667603

SQL Server Interview Questions From Prakash books

http://www.prakashbooks.com/details.php3?id=19008&c=Computer Books

If you want to buy from Amazon

http://www.amazon.co.uk/exec/obidos/ASIN/8183331033/qid%3D1136610981/026-1344994-2263615#product-details

If you want to purchase the book directly through BPB Publication's delhi , India :-

bpb@bol.net or bpb@vsnl.com