Using varchar(MAX) vs TEXT on SQL Server

182

47

I just read that the `VARCHAR(MAX)` datatype (which can store close to 2GB of char data) is the recommended replacement for the `TEXT` datatype in SQL Server 2005 and Next SQL SERVER versions.

If I want to search inside a column for any string, which operation is quicker?

1. Using a the `LIKE` clause against a `VARCHAR(MAX)` column?

```
WHERE COL1 LIKE '%search string%'
```

2. Using the `TEXT` column and put a *Full Text Index*/*Catalog* on this column, and then search using the `CONTAINS` clause?

```
WHERE CONTAINS (Col1, 'MyToken')
```

sql-server performance text varchar sql-types

edited Aug 12 '16 at 9:29
Somnath Muluk
36.2k 17 188 209

asked May 7 '09 at 13:56
user85116
2,047 6 30 31

1 This post is also helpful: stackoverflow.com/questions/564755/… – Jake Jul 8 '10 at 16:55

23 The most important mention in that post is a link to MSDN documentation showing that `TEXT` and `NTEXT` (and `IMAGE`) are deprecated. – Brian May 27 '11 at 19:09

Look at the link: stackoverflow.com/q/28980502/1805776 – vicky Oct 18 '16 at 9:43

5 Answers

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up OR SIGN IN WITH Google Facebook

Using the LIKE statement is identical between the two datatypes. The additional functionality `VARCHAR(MAX)` gives you is that it is also can be used with `=` and `GROUP BY` as any other `VARCHAR` column can be. However, if you do have a lot of data you will have a huge performance issue using these methods.

In regard to if you should use `LIKE` to search, or if you should use *Full Text Indexing* and `CONTAINS`. This question is the same regardless of `VARCHAR(MAX)` or `TEXT`.

If you are searching large amounts of text and performance is key then you should use a *Full Text Index*.

`LIKE` is simpler to implement and is often suitable for small amounts of data, but it has extremely poor performance with large data due to its inability to use an index.

edited Jan 5 '15 at 22:36
potashin
39.2k 9 68 93

answered May 7 '09 at 14:10
Robin Day
82.7k 22 104 156

11 I didn't know that it would store in the page at 8k, and out of the page if larger. Very cool. – Brain2000 May 11 '12 at 18:35

3 Your last line is partially wrong. LIKE can not use index ONLY if wildcard is at the beginning of the string being searched against. – SouravA Jan 5 '15 at 22:42

Is it no problem to alter a field from a text to a varchar(max) from an existing table with data? – user1531040 Feb 17 '17 at 9:23

17

For large text, the *full text index* is *much* faster. But you can *full text index* `varchar(max)` as well.

edited Jan 5 '15 at 22:37
potashin
39.2k 9 68 93

answered May 7 '09 at 13:58
Joel Coehoorn
317k 99 505 740

You can't search a text field without converting it from text to varchar.

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up OR SIGN IN WITH Google Facebook

```
select *
from @table
where a ='a'
```

This give an error:

```
The data types text and varchar are incompatible in the equal to operator.
```

Wheras this does not:

```
declare @table table (a varchar(max))
```

Interestingly, `LIKE` still works, i.e.

```
where a like '%a%'
```

edited Mar 9 '16 at 12:46

rohancragg
4,132 4 29 44

answered May 7 '09 at 14:00

DForck42
10.4k 12 48 75

9 +1 just for saying random downvote! Drives me crazy when people downvote me and have no comment, they really need to get a life. – Tom Stickel Feb 13 '14 at 1:01

3 The reason he got downvotes is that **from what I remember from things I have had to do** is not a valid argument to bring when answering a technical question. Think about people (like me right now) trying to firgure out why should we use `varchar(n)` or `text` , and get over this answer. Do you think, in a professionnal environnement, that arguing with vague statements will help solving the problem ? All the post on StackOverflow are meant to be seen by thousands of people, act in consequences ! – Anwar Nov 2 '15 at 15:20

2 @Zeratops lol, this answer is 6 years old, i was rather green to so when i wrote it. i cleaned up the wording to be more to the point. – DForck42 Nov 3 '15 at 5:12

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up OR SIGN IN WITH Google Facebook

- **Which one to Use?**

As per MSDN link Microsoft is suggesting to avoid using the Text datatype and it will be removed in a future versions of Sql Server. Varchar(Max) is the suggested data type for storing the large string values instead of Text data type.

- **In-Row or Out-of-Row Storage**

Data of a `Text` type column is stored out-of-row in a separate LOB data pages. The row in the table data page will only have a 16 byte pointer to the LOB data page where the actual data is present. While Data of a `Varchar(max)` type column is stored in-row if it is less than or equal to 8000 byte. If Varchar(max) column value is crossing the 8000 bytes then the Varchar(max) column value is stored in a separate LOB data pages and row will only have a 16 byte pointer to the LOB data page where the actual data is present. So `In-Row` Varchar(Max) is good for searches and retrieval.

- **Supported/Unsupported Functionalities**

Some of the string functions, operators or the constructs which doesn't work on the Text type column, but they do work on VarChar(Max) type column.

1. `=` Equal to Operator on VarChar(Max) type column
2. Group by clause on VarChar(Max) type column
    - **System IO Considerations**

As we know that the VarChar(Max) type column values are stored out-of-row only if the length of the value to be stored in it is greater than 8000 bytes or there is not enough space in the row, otherwise it will store it in-row. So if most of the values stored in the VarChar(Max) column are large and stored out-of-row, the data retrieval behavior will almost similar to the one that of the Text type column.

But if most of the values stored in VarChar(Max) type columns are small enough to store in-row. Then retrieval of the data where LOB columns are not included requires the more number of data pages to read as the LOB column value is stored in-row in the same data page where the non-LOB column values are stored. But if the select query includes LOB column then it requires less number of pages to read for the data retrieval compared to the Text type columns.

Conclusion

Use `VarChar(MAX)` data type rather than `TEXT` for good performance.

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up OR SIGN IN WITH Google Facebook


22.4k 10 68 82


36.2k 17 188 209

4

If using *MS Access* (especially older versions like 2003) you are forced to use `TEXT` datatype on SQL Server as MS Access does not recognize `nvarchar(MAX)` as a Memo field in Access, whereas `TEXT` is recognized as a Memo-field.

edited Jan 5 '15 at 22:37


potashin
39.2k 9 68 93

answered Sep 16 '13 at 20:45


Klaus Oberdalhoff
41 2

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up OR SIGN IN WITH Google Facebook