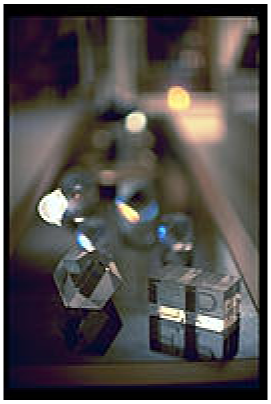
Limits

4 min. read · View original



The most painful limit in Oracle is the 4000-byte maximum length of a VARCHAR. For most Web applications, this turns out to be long enough to hold 97% of user-entered data. The remaining 3% are sufficiently important that you might be unable to use VARCHAR.

Oracle8 includes a Character Large Objects (CLOB) data type, which can be up to 4 GB in size.

```
create table foobar
( mykey integer,
  moby clob );

insert into foobar values ( 1, 'foo');
```

The first time I used CLOBs was for an Oracle 8 port of my Q&A forum software (what you see running at photo.net). I use it for the MESSAGE column. In my Illustra implementation, it turned out that only 46 out of the 12,000 rows in the table had messages longer than 4000 bytes (the

VARCHAR limit in Oracle 8). But those 46 messages were very interesting and sometimes contained quoted material from other works. So it seemed desirable to allow users to post extremely long messages if they want.

Minor Caveats:

- CLOBs don't work like strings. You can't ask for the LENGTH of a CLOB column, for example. You can work around this with PL/SQL calls but it isn't much fun.
- LOBs are not allowed in GROUP BY, ORDER BY, SELECT DISTINCT, aggregates and JOINS
- `alter table bboard modify message clob;` does not work (even with no rows in the table). If you want to use CLOBs, you apparently have to say so at table creation time.

If you thought that these limitations were bad, you haven't gotten to the big one: the Oracle SQL parser can only handle string literals up to 4000 characters in length. SQL*Plus is even more restricted. It can handle strings up to 2500 characters in length. From most Oracle clients, there will in fact be no way to insert data longer than 4000 characters long into a table. A statement/system that works perfectly with a 4000-character string will fail completely with a 4001-character string. You have to completely redesign the way you're doing things to work with strings that *might* be long.

My partner Cotton was custom writing me an Oracle 8 driver for AOLserver (my preferred RDBMS client). So he decided to use C bind variables instead. It turned out that these didn't work either for strings longer than 4000 chars. There is some special CLOB type for C that you could use if you knew in advance that the column was CLOB. But of course Cotton's C code was just taking queries from my AOLserver Tcl code. Without querying the database before every INSERT or SELECT, he had no way of knowing which columns were going to be CLOB.

One of the most heavily touted features of Oracle 8 is that you can partition tables, e.g., say "every row with an order_date column less than January 1, 1997 goes in tablespace A; every row with order_date less than January 1, 1998 goes in tablespace B; all the rest go in tablespace C." Assuming that these tablespaces are all on separate disk drives, this means that when you do a GROUP BY on orders during the last month or two, Oracle isn't sifting through many years worth of data; it only has to scan the partition of the table that resides in tablespace C. Partitioning seems to be a good idea, or at least the Informix customers who've been using it over the years seem to think so. But if you are a CLOB Achiever, you won't be using partitioning.

Right now, my take on CLOBs is that they are so unpleasant to use as to be almost not worth it. It has taken Cotton longer to get this one feature working than everything else he did with his driver. Informix Universal Server lets you have 32,000-character long VARCHARs and I'm looking longingly at that system right now...

Reference

Next: tuning

---

philg@mit.edu

Reader's Comments

> Both MySQL and MS SQL have higher limit of VARCHAR: 1,048,543 and 8000, respectively. This is taken from http://dev.mysql.com/tech-resources/crash-me.php?res_id=38
>
> -- Victor F, August 21, 2004

> A few years ago I was working on a Sybase SQL Server database to store the text of transcribed medical reports. The good folks at Sybase warned us not to use columns of type TEXT (the equivalent of an Oracle CLOB) and to store *lines* of text in a VARCHAR column instead. (At the time Sybase limited VARCHAR columns to 255 bytes.) This works well as long as you don't have any lines longer than 255 characters. The only problem is that you have to parse the text into lines and then do an indefinite number of INSERTs. But it was less work that using the functions to manipulate TEXT data.
>
> -- David Smith, June 29, 2005

> In Oracle 11g length() now works with NCLOBs. Sadly, select distinct still does not.
>
> -- Janek Bogucki, July 2, 2009

Add a comment