Setting NLS LENGTH SEMANTICS

Weird behaviour of DBA client tools with database NLS_LENGTH_SEMANTICS settings

SQL Developer and PL/SQL Developer show some weird behaviour when dealing with a unicode Oracle database.

NLS_LENGTH_SEMANTICS is available as database and session parameter. In a database which is running charset AL32UTF8 or another multibyte charset, this parameter is usually set to CHAR database-wise, which results in new char columns being created with n characters length instead of n bytes length, as it would be with a BYTE setting of this parameter. The default database and session setting is BYTE, unfortunately, obviously for backwards compatibility reasons. See also: Show oracle charset

So if you don't adjust your client and / or database settings and create a new column without specifying the semantics, like:

```
CREATE TABLE emp1
 (empid NUMBER(5),
 fname VARCHAR2(20)
 );
```

you get this table:

```
Name   Null Type
 ----- ---- ------------
 EMPID      NUMBER(5)
 FNAME      VARCHAR2(20)
```

```
select column_name, data_type, char_used from user_tab_columns where table_name = 'EMP';

 COLUMN_NAME  DATA_TYPE  CHAR_USED
 ------------ ---------  ---------
 EMPID        NUMBER
 FNAME        VARCHAR2   B
```

CHAR_USED = B reads as BYTE semantics are used.

You can add 20 single-byte characters in the FNAME column:

```
insert into emp (fname) values ('12345678901234567890');
1 rows inserted.
```

But if you insert multi-byte characters, you get:

```
insert into emp (fname) values ('€€€€€€€€€€€€€€€€€€€€');
 SQL Error: ORA-12899: value too large for column "JSCHREIB"."EMP"."FNAME" (actual: 60, maximum: 20)
```

This teaches us to create tables and their columns with CHAR semantics.
But how can we make sures this is the default for your session ?

First, you can check the database setting. In a unicode database, the parameter ought to be set to CHAR:

```
SQL> alter system set nls_length_semantics = CHAR scope=both;
```

Let's check it in sqlplus:

```
SQL> show parameter sema
 NAME                 TYPE        VALUE
 -------------------- ----------- ---------
 nls_length_semantics string      CHAR
```

looks good.

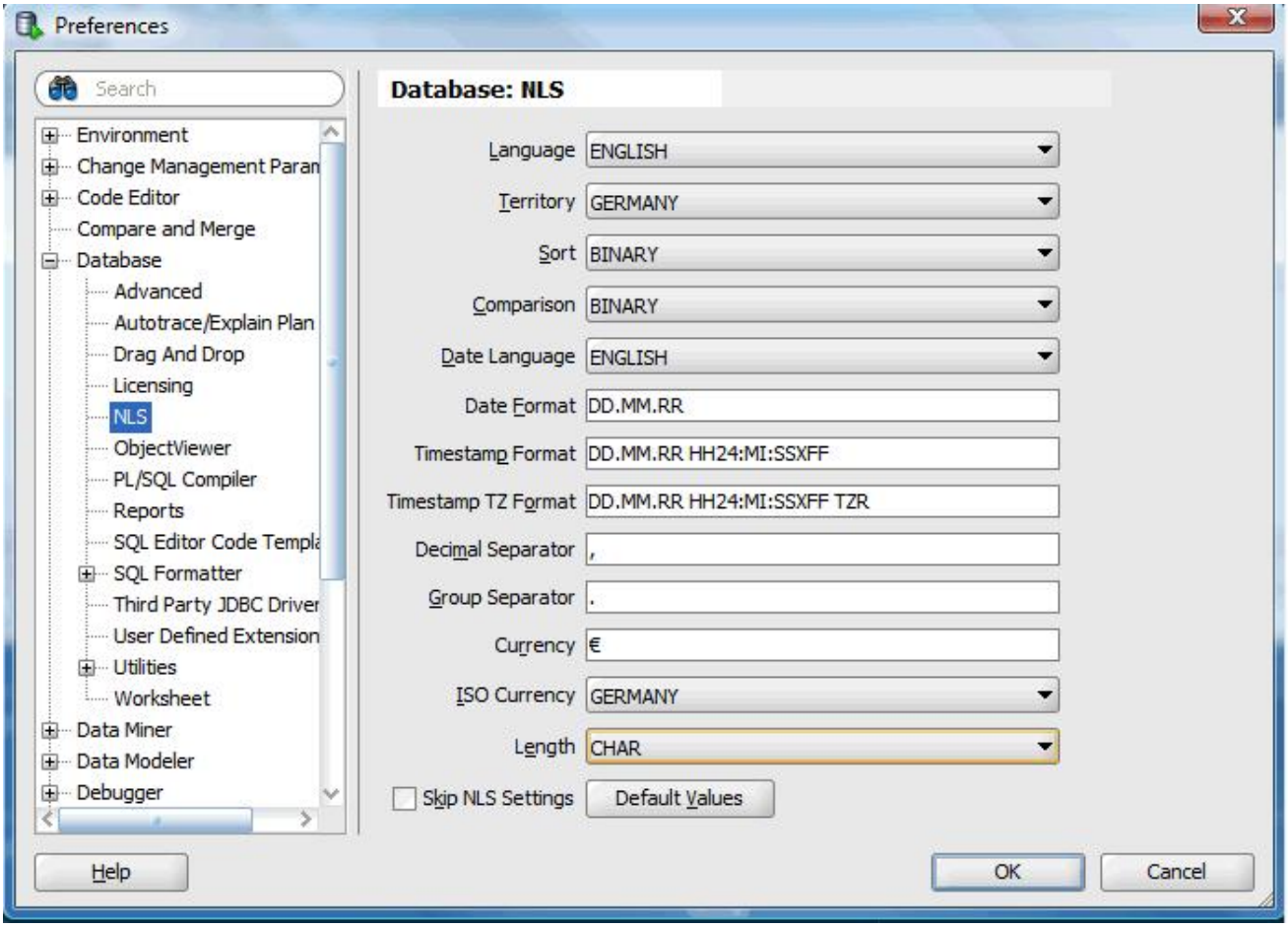
Now, let's check in SQL Developer:

```
select * from v$parameter where name like '%sema%';
select * from v$parameter where name like '%sema%';
select * from v$nls_parameters where Parameter = 'NLS_LENGTH_SEMANTICS';
 PARAMETER VALUE
 ---------------------------------------------------------------
---------------------------------------------------------------
 237 nls_length_semantics BYTE
 237 nls_length_semantics BYTE
 NLS_LENGTH_SEMANTICS BYTE
```

Yes, you are connected to the same database... but SQL Developer **rewrites** the database output!

This is because the tools preferences are set to BYTE.

You can check this in Tools/Preferences/Database/NLS:



Once set to CHAR, SQL Developer has session settings for char semantics and new columns are created in char semantics.
Also, the above query has correct output:

```
select name, value from v$parameter where name like '%sema%';
 nls_length_semantics CHAR
```

In PL/SQL Developer 7.1, the SQL output is correct:

```
select * from v$parameter where name like '%sema%';
 nls_length_semantics 2 CHAR CHAR FALSE TRUE IMMEDIATE
```

But there is a report in "Reports/DBA/NLS Database Parameters", which still states

```
NLS_LENGTH_SEMANTICS BYTE
```

for an unknown reason. Probably a bug, in version 9 history there are a lot of related entries:
[PL/SQL Developer - News|http://www.allroundautomations.com/plsqldevnew.html]