

SQLServer2005与Oracle10g转换方法总结 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/287/2021_2022_SQLServer2_c98_287099.htm 此次需要完成的目标是将库从SQLServer 2005完整的移植到Oracle10g中，包括表结构、数据、视图、函数以及存储过程的移植，移植主要基于Oracle的OMWB（Oracle Migration Workbench）来完成，尽管OMWB能帮助完成大部分具备难度的工作，但还是有很多工作量的事情需要在OMWB完成后来手工进行，所以整个移植过程工作量看起来会非常大，但是不是仅仅只有工作量的问题呢？我觉得不是，写下这篇blog以便需要进行此项操作的同学以及给自己做个备忘。由于目前OMWB仅支持SQLServer2000，根据官方网站的消息，OMWB的下一版会推出对SQLServer 2005的支持，所以在目前的情况下只能先把库从SQLServer 2005移植到SQLServer 2000，这就是我们移植过程的第一步了。

一、SQLServer 2005到SQLServer 2000 一直以来，版本要降级都是很困难的，因为在新版本中必然会有些新的特性，而如果刚好凑巧你使用到了这些特性的话，在降级到低版本时就会碰到一些问题，在经过几次的尝试后，总结而言，这个过程还是比较容易做的，毕竟是同样的数据库，再怎么样也不会出太大的问题，不过也没有像将库从SQLServer 2000升级为SQLServer 2005那么简单，整个移植过程这么进行：1、基于SQLServer 2005的数据导出将表结构和数据导入到SQLServer 2000；这步中需要注意的是默认情况下SQLServer会将表和视图一起导入，在这里不要选择视图，否则导入到SQLServer 2000后有些视图会变成表，选择需要导入的表后基本上这步

不会出现什么问题，可以完成表结构和数据的移植。2、基于SQLServer 2005的生成脚本将视图/函数/存储过程移植到SQLServer 2000；这步需要慢慢来，因为在视图/函数/存储过程中你可能使用到了一些SQLServer 2005的新特性，如果遇到这样的情况，只能是手工进行修改，以使它完全符合SQLServer 2000的要求，尽管在生成脚本时你可以选择生成的目标版本为SQLServer 2000，但还是会有部分脚本执行是会出错的。在完成了SQLServer 2005到SQLServer 2000的移植后，就可以基于OMWB来把库从SQLServer 2000移植到Oracle了，这步尽管有工具，还是会比较的麻烦，总结如下：二

、SQLServer 2000Oracle 10g 关于如何基于OMWB将库从SQLServer 2000移植到Oracle 10g的操作步骤可参见此篇文档：

http://www.oracle.com/technology/global/cn/obe/10gr2_db_vmware/develop/omwb/omwb.htm 大家现在从oracle官方站下的话可能会找不到sqlserver 2000的插件包，如果找不到的话可以从这里下载：我在这里要总结的是基于OMWB将库从SQLServer 2000移植到Oracle 10g后还需要手工做的一些事情，不要指望OMWB能无缝的帮你把库从SQLServer移植到Oracle中，银弹是不存在的，因此我们需要做些手工的工作完成库的移植：

- 1、移植表结构和数据可能会出现的问题；表中字段的默认值/主键/外键/索引移植不过去，这些需要手工的进行补充；
- 2、移植视图可能会出现的问题；移植过去的视图可能会出现各种语法错误的问题，这需要手工的修正，一般来说都是较为简单的错误；另外一种问题就是有些视图可能会无法移植过去，这些视图就只能在对比OMWB的移植报告后找出

来手工的进行移植了。3、移植函数/存储过程可能会出现的问题；移植过去的函数/存储过程中可能仍然会有不少的语法问题，例如像SCOPE_IDENTITY()、REPLICATE、newid()这些OMWB不知道该怎么处理的函数，还有像返回Table类型的这种函数，这些都只能在移植后手工的来进行纠正，关于函数不同造成的语法错误的现象大家可以参看这篇文档来做SQLServer和Oracle函数的对照：

<http://www.mikecat.net/blogview.asp?logID=1559> 移植过去的函数/存储过程可能编译是没有问题，也就是Oracle认为没有语法问题，但执行起来却会报错，像字符串相加，经过OMWB移植后有些字符串相加会替换成||，但是有些会遗漏，这个时候也只能手工来纠正这些错误了；移植过去的函数/存储过程在执行过程中可能会出现某些表的主键值不能为空的现象，造成这种现象的原因多数为在SQLServer中该字段的默认值定义的为IDENTITY，但在Oracle中没法赋予这样的默认值，只能在插入的sql语句中加上对于主键字段的赋值，可采用sequence的方式来生成顺序号；移植过去的函数/存储过程中如果其中的查询语句是采用字符串的方式，然后动态执行的话，这个时候的查询语句就得手工修改为符合oracle的语法了，因为OMWB在移植时是不会对字符串形式的查询语句来做处理的；部分函数/存储过程会由于OMWB确实无法处理，造成移植不到oracle，这个时候也必须参照OMWB的移植报告找出这些函数/存储过程来手工移植了。整个移植过程可能会碰到比上面所列出的更多的问题，可以看出整个移植过程确实需要耗费不小的工作量，但总体而言，完成的难度并不高。其实真的是这样吗？当然不是，就算你完成了上面

的移植工作，那也只能说表面看上去移植是完成了，很有可能会出现这个存储过程语法等等都没有问题了，但执行的效果和SQLServer就是不一样，这是为什么呢？可能会是因为Oracle和SQLServer在并发控制、事务机制上是不同的，而这会影响到程序调用时的sql的编写、存储过程的编写等等，也就是说，在上面的移植过程的工作完成后，还得仔细检查现在的sql语句/函数/存储过程是否根据Oracle的机制达到了原来在SQLServer中期望的效果，只有做到这步的效果是一样的，才可以说移植过程完成了。最后顺带说的就是应该根据Oracle的机制来采用符合oracle优化原则的方法来优化表/视图/函数/存储过程，如果不做这步的话，从sqlserver移植到oracle估计意义也不大了，当然，这可以不列为移植过程的工作。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com