

Oracle10g：数据的导入导出 PDF转换可能丢失图片或格式，
建议阅读原文

https://www.100test.com/kao_ti2020/246/2021_2022_Oracle10g_EF_c102_246110.htm 一、给数据泵加压 Oracle 数据库 10g 中的新的实用程序使其性能和多功能性达到了新的水平。Oracle 数据库 10g 中增加的叫做 Oracle Data Pump（数据泵）的新的导入和导出特性，彻底改变了数据库用户已经习惯的过去几代 Oracle 数据库的客户 / 服务器工作方式。现在服务器可以运行导出和导入任务。你可以通过并行方式快速装入或卸载大量数据，而且你可以在运行过程中调整并行的程度。导出和导入任务现在可以重新启动，所以发生故障不一定意味着要从头开始。API 是公诸于众的，并且易于使用；用 PL/SQL 建立一个导入和导出任务非常简单。一旦启动，这些任务就在后台运行，但你可以通过客户端实用程序从任何地方检查任务的状态和进行修改。二、体系结构在 Oracle 数据库 10g 之前（从 Oracle7 到 Oracle9I），导入和导出实用程序都作为客户端程序运行，并且完成大量工作。导出的数据由数据库实例读出，通过连接传输到导出客户程序，然后写到磁盘上。所有数据在整个导出进程下通过单线程操作。今天的数据量比这个体系结构最初采用的时候要大得多，使得单一导出进程成了一个瓶颈，因为导出任务的性能受限于导出实用程序所能支持的吞吐量。在 Oracle 数据库 10g 和全新的数据泵（Data Pump）体系结构下，如今所有的工作都由数据库实例来完成。数据库实例可以用两种方法来并行处理这些工作：通过建立多个数据泵工作进程来读 / 写正在被导出 / 导入的数据，以及建立并行 I/O 服务器进程以更快地选取（SELECT）

或插入（ INSERT ）这些数据。这样，单进程瓶颈再也就不存在了。数据泵任务用新的 DBMS_DATAPUMP PL/SQL API 来建立、监测和调整。新的导入和导出实用程序（分别为 impdp 和 expdp ）对于这个 API 来说只是命令行接口。你可以使用数据泵导出实用程序初始化一个任务，例如一个导出任务。然后你就可以关闭你的客户端，回家过夜和享用晚餐，而你的任务会一直运行。到了深夜，你可以重新连接到那个任务，检查其状态，甚至可以提高并行程度，以便在深夜系统没有用户在用的情况下多完成一些工作。第二天早上，你可以降低并行度甚至挂起该任务，为白天在线的用户释放资源。重新启动任务的功能是数据泵体系结构的一个重要特性。你可以随时停止和重新启动一个数据泵任务，比如为在线用户释放资源。你还可以从文件系统的空间问题中轻松地恢复。如果一个 12 小时的导出任务在进行了 11 小时后因磁盘空间不够而失败，那么你再也不用从头开始重新启动该任务，重复前面 11 小时的工作。而是你可以连接到这个失败的任务，增加一个或多个新的转储（ dump ）文件，从失败的地方重新启动，这样只需一个小时你就可以完成任务了。这在你处理很大数据量时非常有用。

三、对文件系统的访问

由服务器处理所有的文件 I/O 对于远程执行导出和导入任务的数据库管理员来说非常有利。如今，用户可以很轻松地在类似 UNIX 的系统（如 Linux）上 telnet 或 ssh 到一个服务器，在命令行方式下初始化一个运行在服务器上的导出或导入任务。然而，在其他操作系统上就不那么容易，Windows 是最明显的例子。在推出数据泵之前，要从一个 Windows 系统下的 Oracle 数据库中导出大量数据，你很可能必须坐在服务器控

制台前发出命令。通过 TCP/IP 连接导出数据只对小数据量是可行的。数据泵改变了这一切，因为即使你通过在你的客户端上运行该导出和导入实用程序来初始化一个导出或导入任务，该任务其实也运行在服务器上，所有的 I/O 也都发生在该服务器上。出于安全性考虑，数据泵要求你通过 Oracle 的目录对象来指定其中存放着你要建立或读取的转储文件的目标目录。例如：CREATE DIRECTORY export_dumps AS c:\a. GRANT read, write ON DIRECTORY export_dumps TO gennick. 我以 SYSTEM 身份登录到我的实验室数据库上，并执行以上语句来建立一个目录对象，这个目录对象指向了我磁盘上的一个临时目录，以用来存放导出的转储文件。GRANT 语句为用户 gennick- 就是我 - 分配了访问该目录的权限。我给自己分配读 / 写权限，因为我将导出和导入数据。你可以为一个用户分配读权限，限制他只能导入数据。

四、启动一个导出任务 你可以使用新的 expdp 实用程序来启动一个导出任务。因为参数与老的 exp 实用程序不同，所以你得熟悉这些新的参数。你可以在命令行中指定参数，但在本文中我使用了参数文件。我想导出我的整个模式（ schema ），使用了以下参数： DUMPFILE=gnis%U.dmp DIRECTORY=export_dumps LOGFILE=gnis_export.log JOB_NAME=gnis_export DUMPFILE 指定我将向其中写入被导出数据的文件。 %U 语法给出了一个增量计数器，得到文件名 gnis01.dmp、 gnis02.dmp 等。 DIRECTORY 指定了我的目标目录。 我的 LOGFILE 参数指定了日志文件的名称，这个文件是为每个导出任务默认创建的。 JOB_NAME 给任务指定了一个名字。我选择了一个易于记忆（和输入）的名字，因为我可能需要在后面才连接这个任

务。要注意在指定任务名称时不要与你登录模式（ schema ）中的模式对象名称冲突。数据泵在你的登录模式中建立一个被称为任务主表的数据表，该表的名字与任务的名字相匹配。这个数据表跟踪该任务的状态，并最终被写入转储文件中，作为该文件所含内容的一个记录。清单 1 显示了一个导出任务已被启动。该任务所做的第一件事是估计所需的磁盘空间大小。当估计值显示出来后，我按 ctrl-C 进入一个交互式的导出提示窗口，然后使用 EXIT_CLIENT 命令回到我操作系统的命令窗口。该导出任务仍然运行在服务器上。注意，如果我要做并行导出并且将我的 I/O 分布在两个磁盘上，那么我可以对 DUMPFILE 参数值做出修改，并如下添加 PARALLEL 参数和值，如下所示：

```
DUMPFILE=export_dumps01:gnis%U.dmp,  
export_dumps02:gnis%U.dmp PARALLEL=2
```

注意，在这个并行导出任务中，目录名作为文件名的一部分来被指定。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