# Generating Parameter Files for 9i Exports to Refresh Azure Databases

## Introduction

The first test of importing a database export, of depersonalised data from production, took over 89 hours to complete. This was initially traced to a couple of large tables taking around 24 hours to import and update the indexes which existed at the time of the import.

Subsequent investigation, and numerous full tests allowed the import time to be reduced to around 24 hours with 7 separate import processes running concurrently.

## The Problem

The problem with concurrent imports is that each one needs a separate export file and the list of tables must be hard coded in the *tables=(…)* parameter. If any new tables are created then these must be added to an existing, or a new parameter file otherwise an incomplete import may be the result.

## The Solution

There are a number of scripts supplied that will:

* Create a package and package body named *sys.xxnd\_parfiles*.
* Execute various procedures in the above package to generate the required parfiles in the current directory.
* Drop the above package after use.

The scripts are:

* XXND\_PARFILES.package.sql - creates the package.
* XXND\_PARFILES.package\_body.sql - creates the package body.
* Generate\_parfiles.sql - runs the procedures in the above package to generate the required parfiles for a full, parallel export of the production database. The following users are exported:
  + CMTEMP
  + FCS
  + ITOPS
  + LEEDS\_CONFIG
  + OEIC\_RECALC
  + ONLOAD
  + UVSCHEDULER
  + Plus any other users who's account status is not "EXPIRED & LOCKED" and which owns objects.
* Drop.package.XXND\_PARFILES.sql - drops the above package after use.

## Running the Code

The code should be run on the production database standby either before or after depersonalisation, as per security rules.

* cd to a suitable directory on the server where the parameter files will be created.
* Using SQL\*Plus, login to the database as SYSDBA.
* Execute the script XXND\_PARFILES.package.sql to create the package.
* Execute the script XXND\_PARFILES.package\_body.sql to create the package body.
* Execute the script generate\_parfiles. You should be prompted for a location for the '*Output\_directory\_for\_dumpfiles*' - enter the location where the exp process will create the dumpfiles and logfiles. All the generated parfiles will use this location for their dump and log files at execution time.
* Wait - it took around 2-3 minutes in testing.
* Check that the following files have been generated and contain valid content:
  + exp\_NOROWS.par
  + exp\_ROWS\_NOFCS.par
  + exp\_ROWS\_FCS1.par
  + exp\_ROWS\_FCS2D.par
  + exp\_ROWS\_FCS3.par
  + exp\_ROWS\_FCS4.par
  + exp\_ROWS\_FCS5.par
  + exp\_ROWS\_FCS6.par
  + exp\_ROWS\_FCS7.par
  + exp\_ROWS\_FCS8.par – **Note:** this one will have a table name in MiXeD case characters, this is correct. **Do not adjust**. In addition the name will be wrapped in three sets of double quotes. This is also correct. **Do not adjust**.
  + exp\_ROWS\_FCS9.par
* If happy, the script drop.package.XXND\_PARFILES.sql can be run to drop the package.

## Running the Exports

The generated parameter files should be used as follows to run the parallel exports:

* cd to a suitable location on the server.
* Set the Oracle environment as normal.
* Run the NOROWS export first:
  + exp sys/password parfile=exp\_NOROWS.par
* Run the remaining exports in parallel, in Unix background mode:
  + exp sys/password parfile=exp\_NOFCS.par &
  + exp sys/password parfile=exp\_FCS1.par &
  + exp sys/password parfile=exp\_FCS2D.par &
  + exp sys/password parfile=exp\_FCS3.par &
  + exp sys/password parfile=exp\_FCS4.par &
  + exp sys/password parfile=exp\_FCS5.par &
  + exp sys/password parfile=exp\_FCS6.par &
  + exp sys/password parfile=exp\_FCS7.par &
  + exp sys/password parfile=exp\_FCS8.par &
  + exp sys/password parfile=exp\_FCS9.par &
* When complete, the log files should be checked for errors. They are created in the same location as the dump files, and this is the location you were prompted for earlier when generating the parfiles.
* Zip up the various dump and log files.
* (S)FTP to a suitable location on the Azure servers, or, copy to a location that the Leeds DBA Team can access and we will copy the files to Azure.

## Any Questions?

Any problems or questions, suggestions for improvements? Contact [norman.dunbar@capita.co.uk](mailto:norman.dunbar@capita.co.uk).