

# Coldfusion and DTS - Easy Exporting

Posted At : July 26, 2006 4:43 PM | Posted By : Mark Kruger

Related Categories: MS SQL Server, Coldfusion Tips and Techniques

What do you do when you are required to provide a CSV export of a large dataset from MS SQL Server? You could use query2csv and export it in Coldfusion - but don't be surprised if you end up taking a long time to complete that request. Coldfusion, for all it's advantages, it is not suited to to this sort of thing. We had a process that exported 30,000+ records (just a few fields) for the purpose of sales calculations. In Coldfusion this resulted in a 6 meg file - that doesn't sound like much, but the process could take 5 minutes or more. We thought of DTS, but one of our requirements was to make the file accessible via FTP. Fortunately there was an easy way.

In short, we used a DTS package on the CF Server to create the file locally with DTS services. We shortened the time from 5 minutes to about 35 seconds (not bad). Here are the steps.

## Step 1: Create the DTS on the MS SQL Server

Go through the process of creating the DTS. This might involve simply creating a query and using it in the DTS or it might involved a stored procedure, view or whatever - the main thing is that the DTS uses a query to export to a "flat file". The flat file option will ask you for a location. The location will be on the local drive (where you are running it from). Then change the path in step 2.

## Step 2: Change the Path in the DTS Package

Determine the correct path of the file *on the web server* and enter that path into the package using the package designer. Note: You will not be able to test this step until you run the DTS package from the CF server.

## Step 3: The Coldfusion Code

You will now write code that resembles this sample.

```
<cfscript>
    create dts package object
pkg = createObject("COM", "DTS.Package");
    // load package
pkg.LoadfromSQLServer
    ("NameOfSQLServer",
    "SQL_Username",
    "sql_password",
    0,
    "",
    "",
    "",
    "nameOfTheDtsPackage",
    "");
    // execute
pkg.Execute();
</cfscript>
```

If your path is correct in the package you will be able to look in the designated folder and find the newly created file.

## DTS's Other Uses

Incidentally, I've found DTS to be useful for moving data around outside of MS SQL. For example, with enterprise manager and 2 ODBC connections to 2 different MySQL servers you can migrate data back and forth willy nilly. Or how about from dbase to a flat file? Or from a flat file to Oracle? It's easy to forget that DTS is designed to translate data between a number of different platforms and file types - not just MS SQL.