



Custom Query Extracts

Custom Query Extracts

PS_QUERY

Application 14773

DNAApp Id a442e44f-1d61-4148-9c5b-b70490d56949

Fiserv Confidential: Distribution restricted to:

- Clients using or considering purchase of the product described in this document
- Fiserv associates

© 2012-2019 Fiserv, Inc. or its affiliates. All rights reserved. This work is confidential and its use is strictly limited. Use is permitted only in accordance with the terms of the agreement under which it was furnished. Any other use, duplication, or dissemination without the prior written consent of Fiserv, Inc. or its affiliates is strictly prohibited. The information contained herein is subject to change without notice. Except as specified by the agreement under which the materials are furnished, Fiserv, Inc. and its affiliates do not accept any liabilities with respect to the information contained herein and is not responsible for any direct, indirect, special, consequential or exemplary damages resulting from the use of this information. No warranties, either express or implied, are granted or extended by this document.

<http://www.fiserv.com>

Fiserv is a registered trademark of Fiserv, Inc.

Other brands and their products are trademarks or registered trademarks of their respective holders and should be noted as such.

Overview:

The Custom Query Extracts gives financial institutions the ability to automate the production of delimited files from their own SQL queries.

Key Benefits:

This application can improve efficiency through automation.

Processing:

The PS_QUERY application consumes a text file containing one or more SQL queries, as described in the *File Layouts* section of this document and creates a delimited file with the query results. Runtime parameters are provided to specify the input file path and input file name, as well as the desired delimiter type and output file destination.

All extract files created from valid SQL queries are placed in the path specified by the “Output File Path” (COUP) parameter. Please note, if you have the “Output File Path” (COUP) parameter set to a specific folder across multiple runs with the same input file, each run will overwrite the extract file created by the previous.

Each SQL query is processed individually. If one SQL query has a bug, an error is reported and processing continues with the next SQL query in the file. This process does not support updating the database. Only read only queries are allowed.

More than one instance of this batch application should not be run simultaneously.

Parameters:

This application uses the following runtime parameters:

Parameter	Code	Description (how used)	Required	Default
Input File Path	IPTH	The input file path of the SQL queries text file. e.g. “C:\PS_QUERY\INPUT”	Yes	<Blank>
Input File Name	IFLE	The input file name of the SQL queries text file. e.g. “PS_QUERY.TXT”	Yes	<Blank>
Output File Path	COUP	The output file path where the delimited files should be created. The output file name is specified per SQL query inside the input file. e.g. “C:\PS_QUERY\OUTPUT”	No	Batch Queue Output Directory
Delimiter Type	8DLM	The delimiter to use with the input file. Each SQL query in the input file will return results delimited by the character, or characters that are specified in this parameter. To create tab delimited output, use the word TAB	Yes	,

EXTN Schema:

This application creates one external table as PS_QUERY. This tables stores the query number and output result of the query. This table is used to write the records in CSV output file and deleted after the output file is complete.

PS_QUERY

Name	Description (how used)	Data Type	Default
NBR	The number of output record	Number (22)	<Blank>
CSV	The output record of the query	VARCHAR2 (4000)	<Blank>

Report:

PS_QUERY produces a report containing the following details:

Field	Description
Output File Name	Output File Name is the name of the output file found in the input file.
SQL Description	SQL Description is the description of the SQL found in the input file.
Record Count	Record Count is the total count of records returned by the SQL.
Query Time	Query Time is the total amount of time (in seconds) it took the database server to execute the SQL query.
File Time	File Time is the total amount of time (in seconds) it took the batch workstation to download and write the records to the extract file.

Sample Report

```

Bank: TEST INSTITUTION                               Custom Query Extracts                               Run Date: 03-27-2014
Report: PS_QUERY                                     Post Date: 03-26-2014

Queue Number      : 34287                               Run Time: 09:29:04
Application Number: 14773                               Cash Box:
Queue Sub Number  : 2

                                SCHEMA
                                OSIBANK

                                DATABASE NAME
                                CS8.WORLD

                                RELEASE
                                DNA 3.3.0.3

                                09-26-2012 12:02:28 PM 195121 G:\OSI\BANK\BAT_EXE\EXTNS\

                                REPORT PARAMETERS

                                Delimiter Type: ,
                                Output File Path: G:\OUTPUTFILE\
                                Input File Name: PS_QUERY_INPUT.SQL
                                Input File Path: G:\INPUTFILE\

Bank: TEST INSTITUTION                               Custom Query Extracts                               Run Date: 03-27-2014
Report: PS_QUERY                                     Post Date: 03-26-2014

Output File Name      SQL Description      Record Count      Query Time      File Time
ACCTS_PER_PRODUCT.CSV A count OF accounts per product.      548              1              1
BIRTHDAYS.CSV        Tomorrow's employee birthdays.         0                0              0

Bank: TEST INSTITUTION                               Custom Query Extracts                               Run Date: 03-27-2014
Report: PS_QUERY                                     Post Date: 03-26-2014
Page: 2 of 2

System Report Information: Queue Number = 34287      Application Number = 14773      Queue Sub Number = 2

Account Number      Activity Error Message      Oracle Error Message
0                   BIRTHDAYS.CSV              ORA-00904: "PE"."EFDATE": invalid identifier
    
```

File Layout:**Input File Layout:**

Each SQL query request in the input file requires three parts in the following order:

1. Output File Name (identified by a colon, followed by a valid file name)
2. SQL Description (identified by two hyphens)
3. SQL Query (all text starting on the line after the SQL Query Description, ending with the line before the next Output File Name or the end of the file.

For the convenience of the user writing the SQL queries, the following variables are provided:

- [TODAY] = the current effective date of the queue
- [MONTH] = the first day of the month based on the effective date of the queue
- [QTR] = the first day of the quarter based on the effective date of the queue
- [YEAR] = the first day of the year based on the effective date of the queue
- [Q1] = Date range of the current year's first quarter start date AND end date (Shall be used with BETWEEN function)
- [Q2] = Date range of the current year's second quarter start date AND end date (Shall be used with BETWEEN function)
- [Q3] = Date range of the current year's third quarter start date AND end date (Shall be used with BETWEEN function)
- [Q4] = Date range of the current year's fourth quarter start date AND end date (Shall be used with BETWEEN function)

Restrictions in Input File

- SQL query cannot contain DDL commands.
- SQL queries cannot update the database.
- SQL query cannot exceed 4000 characters in length.
- Result set cannot exceed 4000 characters per record, including the delimiter.
- The Output File Name and SQL Description do not have a length limit set by this application, but only the first 40 characters of each will appear on the report.
- If more than one SQL Description is found, the last one will found will be used.
- The Output File Name and SQL Description identifiers must be in the first position.

Input File Example

```

:ACCTS_PER_PRODUCT.CSV
--A count OF accounts per product.
SELECT
A.MjAcctTypCd Major,
A.CurrMiAcctTypCd Minor,
A.CurrAcctStatCd Status,
COUNT( A.AcctNbr) Total
FROM
Acct A
GROUP BY
A.MjAcctTypCd,
A.CurrMiAcctTypCd,
A.CurrAcctStatCd

:BIRTHDAYS.CSV
--Tomorrow's employee birthdays.
SELECT
P.LastName,
P.FirstName,
EXTRACT(YEAR FROM [TODAY]) - EXTRACT(YEAR FROM P.DateBirth) Age,
EXTRACT(YEAR FROM [TODAY]) - EXTRACT(YEAR FROM PE.EfDate) Loyalty
FROM
Pers P
JOIN PersEmpl PE
ON PE.PersNbr = P.PersNbr
AND InactiveDate IS NULL
WHERE
EXTRACT(MONTH FROM P.DateBirth) = EXTRACT(MONTH FROM [TODAY] + 1)
AND EXTRACT(DAY FROM P.DateBirth) = EXTRACT(DAY FROM [TODAY] + 1)
    
```

Configuration Checklist:

Item	Test Environment	Production Environment
Parameters		
Input File		

Installation:

Note: If you obtained this application from the DNAAppstore, please disregard the installation instructions below. If you did not obtain this application from the DNAAppstore, please complete the installation instructions below.

- Run PS_QUERY_DDL.SQL using an SQL tool such as SQLPlus
- Load the PACK_PS_QUERY_WRAP.SQL, PACK_PS_QUERY_BODY_WRAP.SQL custom stored procedure using Oracle Object Manager.
- Run the PS_QUERY.SQL with an SQL tool like SQLTalk.
- Copy the PS_QUERY.SQT to the PS Batch Report Directory, as specified by the B ATP Institution Option.
- This application requires an Authorization Key. After the primary installation instructions have been completed, you must apply the Authorization Key. Using the System Manager/Application Manager functionality, enter the supplied Authorization Key for this application.

This application includes a custom stored procedure that needs to be loaded on the database. As with any stored procedure, please do not load it while the system is in ONLINE mode, or while batch is running. To load the stored procedures please use the Oracle Object Manager in the SAF Resource Kit.

To download Resource Kit: On the Extranet, go to Core > SAF > SAF Resource Kit Releases > Latest Release > Download File. This will install the resource kit.

To run the utilities: Start > All Programs > Open Solutions Resource Kit will open windows explorer showing various utilities. Open OracleObjectManager and launch OSI.SafRk.OracleObjectManager.exe.

The SAF Resource Kit tools use the Oracle client and therefore should preferably be run from the App Server, or at least from a machine that has the Oracle client installed on it.

If you have any questions while using these tools, please contact the Client Care group for assistance.

Revisions:

Date	App Version #	Change
08/2020	1.0.0.1	Changed label to Fiserv Confidential
12/2019	1.0.0.1	Added tab delimited output functionality.
02/2018	1.0.0.0	Documentation update - Added Fiserv Confidential Label
02/2016	1.0.0.0	Documentation Format Updated
07/2012	1.0.0.0	Re-packaged and Re-documented for DNA™ AppStore
03/2012	1.0.0.0	Application Created