CAL POLY POMONA UNIVERSITY LIBRARY

CFS LIBRARY INTERFACE

**OVERVIEW**

The purpose of the CFS Library Interface is to take payment data that has been entered into the Millennium and output it to CFS so that checks can be produced for the vendors.

**Main Components**

* chargeact.out - output file created by INNOPAC
* lib2cfs.pl – perl script that modifies the chargeact.out data into the format acceptable by CFS

**Millennium Vendor Record Special Notes**

* external fund code structure (length - 20 characters)
	+ account number – columns 1-6
	+ fund number – columns 7-11
	+ department number – columns 12-16
	+ program number – columns 17-20

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Column | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| Coding | **account** | **fund** | **department** | **program** |
| Examples | 6 | 0 | 8 | 0 | 0 | 1 | P | O | M | 0 | 1 | 5 | 5 | 8 | 0 | 0 | 0 | 4 | 0 | 1 |
| 6 | 0 | 8 | 0 | 0 | 3 | T | Y | 0 | 3 | 3 | 5 | 5 | 8 | 0 | 0 | 0 | 4 | 0 | 1 |

* alternate vendor code (d) (length - 10 characters)
	+ PeopleSoft vendor code**\*** – column 1-8
	+ Address sequence # - column 9-10

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Column | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Coding | **people soft vendor code with leading 0** | **addr seq # - with leading 0** |
| Examples | 0 | 0 | 0 | 0 | 1 | 0 | 4 | 1 | 0 | 1 |
| 0 | 0 | 0 | 1 | 3 | 0 | 2 | 8 | 0 | 2 |

\* Under the assumption that the PeopleSoft Vendor Code does not exceed 8 digits. (Need to update the Perl program when the PeopleSoft Vendor Code is 9 digits long.)

**CFS Data File Corresponding Fields**

|  |  |  |
| --- | --- | --- |
| **Column**  | **Header** | **Millennium Field Name or value** |
| A | Invoice ID | Invoice number |
| B | Invoice Dt | Inv date |
| C | Vendor ID | Alt vendor code column 1 - 8 padded with 2 leading 0 to satisfy the 10 characters length requirement |
| D | Acctg Dt | Paid date |
| E | Vchr Line # | 1 |
| I | Gross Amt | Amount + Tax + Ship + Discount |
| J | Merchandise Amt | Amount |
| K | Descr | Vendor name |
| M | Account | External fund column 1 - 6 |
| N | Fund | External fund column 7 – 11 |
| O | Dept | External fund column 12 – 16 |
| P | Program | External fund column 17 -20 |
| Q | Class | C3166 for TY033 fund, 00000 for POM01 fund **\*\*** |
| S | Ship to Location | 75 when SUT = S or U, NT when SUT = E |
| T | SUT | S if Tax ≠ blank, U if Use tax ≠ blank, E if no Tax & Use tax |
| U | Sales Tax | Tax  |
| V | Freight Amt | Ship |
| W | Misc Amt | Discount |
| AC | Addr Seq # | Alt vendor code column 9 - 10 |

(Detail information of Millennium Fields is in the Innovative Millennium Guide Page #105988)

\*\* Fund – Class tables

LOTTERY

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Account | Fund | Department | Program | Class |
| 608001 | TY033 | 55800 | 0401 | C3166 |
| 608003 | TY033 | 55800 | 0401 | C3166 |
| 608004 | TY033 | 55800 | 0401 | C3166 |
| 608005 | TY033 | 55800 | 0401 | C3166 |
| 608806 | TY033 | 55800 | 0401 | C3166 |

STATE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Account | Fund | Department | Program | Class |
| 608001 | POM01 | 55800 | 0401 | 00000 |
| 608003 | POM01 | 55800 | 0401 | 00000 |
| 608004 | POM01 | 55800 | 0401 | 00000 |

**INSTALLATION**

* Install ActivePerl onto the PC (<http://www.activestate.com/activeperl/downloads>).
	+ The script lib2cfs.pl was last tested under ActivePerl 5.12.3 (64-bit)
	+ When installing ActivePerl, be sure to say yes to the option to add c:\perl\bin to the path
* Create a folder to run the CFS interface script, e.g., […]\cppcfs
* Copy the script lib2cfs.pl to this folder

**PROCEDURES**

1. Save the Millennium Output Voucher File to the designated folder, e.g., \cppcfs, and keep the default file name, **chargeact.out**
2. Double click the perl script **lib2cfs.pl** to start the interface program
3. A log file will be created with current month and day in the file name; e.g., lib2cfslog0309.txt will be created when running the program on March 9. Check this file to verify if there is any run time problem. Below are some sample logs
	1. Wed Mar 9 15:03:23 2011
	chargeact.out file is not in the directory
	2. Wed Mar 9 15:08:28 2011
	Can't create today's data folder (20110309): File exists
	3. Wed Mar 9 14:56:09 2011

Getting INNOPAC data...please wait.

Got data. There are 137 records in the input file.

The last record is valid.

Removing unwanted records...please wait.

Now 117 records to process.

Processing...please wait.

DONE!

1. A subfolder contains the following output files will be created using today’s date (*yyyymmdd*) as the folder name; e.g., 20110309 when running the interface program on March 9, 2011.
	1. chargeact*mmdd*  (Millennium input file chargeact.out renamed; e.g., chargeact0309)
	2. errors*mmdd*.txt (list vouchers that the interface program cannot process; e.g., errors0309.txt)

Wed Mar 9 14:56:09 2011

The following vouchers were not processed:

============================================================

Voucher: Reason

============================================================

60819 : Blank space(s) in the PeopleSoft vendor location code. Check the Vendor Record d field.

60820 : Blank space(s) in the PeopleSoft vendor location code. Check the Vendor Record d field.

60821 : Blank space(s) in the PeopleSoft vendor location code. Check the Vendor Record d field.

* 1. **vouchers*mmdd*.csv** (output file for CFS; e.g., vouchers0309.csv)

**ACKNOWLEDGEMENT**

The original Perl Script was posted on the IUG Clearinghouse by Phil Youngholm from MARINet (<http://www.innovativeusers.org/iug_clearinghouse-app/index.php?qfdo=Full+Record&id=55>)

Last updated: April 11, 2011 Library Systems