KU Report Process Creation Agent & Output

General Summary of Function

Agent in AgentTester local DB that I use for running agents and passing in data for configuration settings. Name of the document in AgentTester is KU\_Report\_n\_Process. Purpose of the agent is to go through an input file that was provided take the pieces of that file and determine what pieces in eReg exist or do not exist so that scenarios can be created and then coded for dealing with offboarding or KU’s. The pieces that will be determined will be NCOUAR DB Active/Deleted, NCOUAR UK DB Active/Deleted, NAB, & Freeze NAB. Then taking those pieces look for matches of the various DB’s and the input data etc. Finally, a check of Blue Pages will also be done. All data will be written out to an output file to match up with the input file.

Once the output file is created architect can use that data in the determination of the scenarios that need to be addressed to off board the users accounts and email files.

Codebase Details

Agent: Modules as they are flowed into via Initialize Routine.

loadUIDocFields – Pulls in the configuration data from the agent document in AgentTester DB.

setupDefaults – Sets the default values for various data objects such as DB’s and Views used later in code.

openInputCSVFile – Takes file name for input from configuration settings and opens it.

setupOutputStream – Creates the output file to be written to.

writeHeaderRecord – For the Excel output file the header row is written here

Input file is then read in one line at time and looped through via the following modules and lines of code.

Each line is split out into an array by a comma delimiter.

We are expecting 6 columns of data via input file. Check column count, if not a match then log/skip record.

With each matched record

loadOutputFields – loads an array with the values from the input line of data.

ProcessRecord – takes each line from input file and processes it

Sets up the serial number to make sure it is 9 digits in length.

Sets up the input columns of data into fields to be used in next set of modules.

searchNCOUARDB – searches active/deleted views for both NCOUAR & NCOUAR UK

setNCOUARFields – sets the fields for output from the searches.

setNCOUARUKFields – sets the fields for output from the searches.

searchNABDB – searches NAB & Freeze NAB

searchMatches – Took a formula from NAB DB and converted it to Lotus Script and checks all the

various matches that we are looking for in our output.

processBluePages – Uses the Blue Pages Script Library to check and see if we get a return code on

by serial number lookup into Blue Pages SDK.

writeOutputRecords – Takes all the fields and searches found during processRecords and writes them

into an output row.

cleanOutFields – Resets all the fields for the next record.

End of loop for each record/line in the file.

Close out the output Stream

Output Report

1. Columns 1-6 (Original Input File)
   1. Country
   2. Date
   3. Name (I removed commas here for those in last, first)
   4. Email
   5. Mail Server
   6. Input UID
2. Processed Output
   1. Padded Serial Number (Leading Zeros to 9 places with a ‘ put in front to keep zeros in Excel)
   2. NAB Email Address – Based on formula used in Views converted to Lotus Script
   3. NCOUAR DB Active by Serial Number
   4. NCOUAR DB Deleted by Serial Number
   5. NCOUAR UK DB Active by Serial Number
   6. NCOUAR UK DB Deleted by Serial Number
   7. NCOUAR DB Short Name
   8. NCOUAR UK DB Short Name
   9. NAB Short Name
   10. Freeze NAB Short Name
   11. NCOUAR DB Domain Name
   12. NCOUAR UK DB Domain Name
   13. NCOUAR DB Serial Number
   14. NCOUAR UK DB Serial Number
   15. NAB DB By Short Name
   16. Freeze NAB DB by Short Name
   17. Match between input file and NAB
   18. Match between input serial # and NCOUAR Serial #
   19. Match between input serial # and NCOUAR UK Serial #
   20. Is there only 1 record in NCOUAR/NCOUARUK DB?
   21. Match between NAB & NCOUAR DB Short Name
   22. Match between NAB & NCOUAR UK DB Short Name
   23. Bluepages Return Code
   24. Bluepages Return String