**Message manager 2 steps:**

**Part I: These steps are designed to handle up to 2 message areas with *fixed positions*. To set up *free-flow message*, please see Part II of this document.**

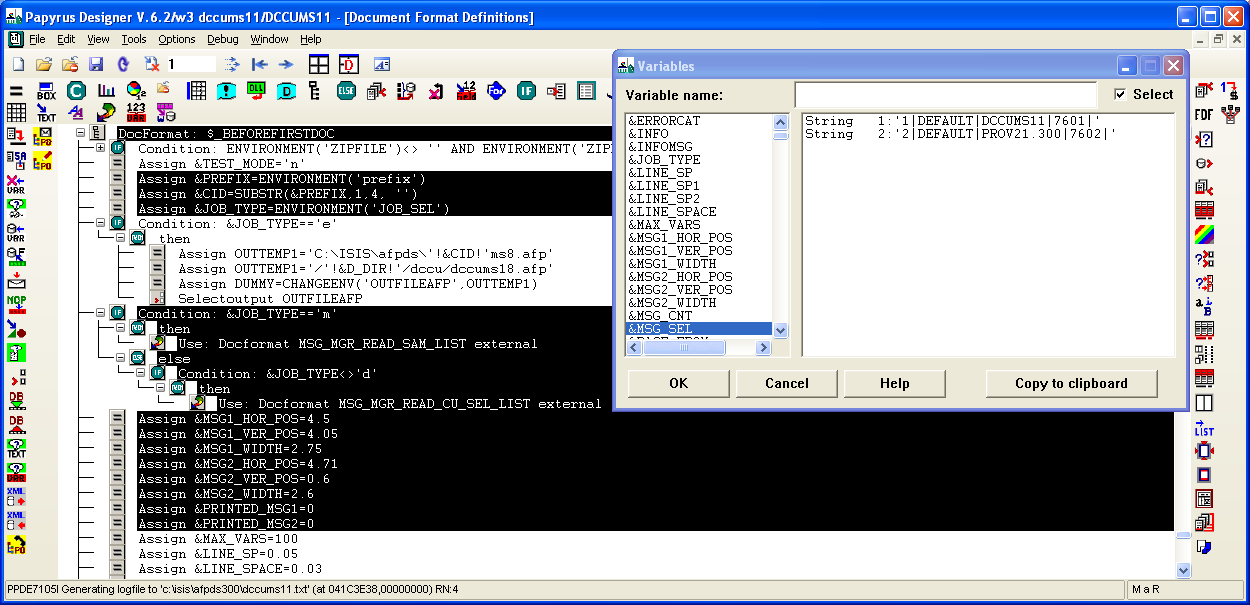
Step 0: Provide ISD the message information such as: width, height, font name, font size, maximum number of lines, maximum characters, etc.

* NOTE: Here are some model programs (by data core) you can use as a reference. Pick one that most reflects your data core.

1. PRINT IMAGE - OSCUMS11.dfa, OSCUMV11.dfa, NTSBLS11.dfa
2. OSI – PROVMS11.dfa, SVBTMS11.dfa
3. ULTRA DATA – DCCUMS11.dfa
4. SYMITAR DATA – NMCUMS11.dfa
5. MISER – GECUMS11.dfa

Step 1: Set up global variables such as the horizontal and vertical positions of the message, and store selective criteria entries from the index file ($prefix\_msg\_portal.txt) in &MSG\_SEL array.

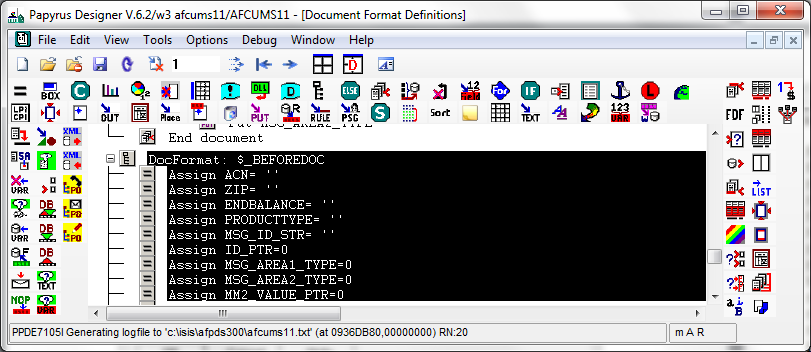
* NOTE: The coding below should be added to BEFOREFIRSTDOC Docformat.



* /home/keep/msgmgr2\_sample\_portal.txt is an example of how an index file with selective criteria looks like. You can rename this file using the specified naming convention ($prefix\_msg\_portal.txt), and modify it accordingly for testing purpose.

Step 2: Initialize local variables such as selective criteria name (e.g. ACN = selective by account number, ZIP = selective by zip code, etc.)

* NOTE: The coding below should be added to BEFOREDOC Docformat.



NOTE: \*\*\*SKIP THIS STEP 3 IF YOU ARE SETTING UP MESSAGE MANAGER 1.0 – NO SELECTIVE MESSAGING\*\*\*\*  
Step 3: Extract/Calculate and store selective criteria information in a predefined table (see below for the table’s definition) called MM2\_VALUE\_TABLE indexed by MM2\_VALUE\_PTR **(Please be sure to use these names)**. Table is only needed if the criteria have more than 1 possible value such as product type. For example, a product type can be of Checking, Savings, CD, or Money Market, etc. If ALL your selective criteria have single value, please do the following:

1. Assign MM2\_VALUE\_PTR = 1 (done at BEFOREDOC docformat)
2. Assign MM2\_VALUE\_TABLE[1] = ‘’ (empty) (done at BEFOREDOC docformat)

-Definition for MM2\_VALUE\_TABLE: (It is pipe ‘|’ delimited)

**\*\*IMPORTANT\*\***: If there is any selective criteria that CANNOT be found in the list below, please let Tien know to add it to the table.

Field 1) PRODUCTTYPE

Field 2) ENDBALANCE/ENDINGBAL

Field 3) MATURITYDAYS

Field 4) APPLICATIONCODE

Field 5) MAJOR

Field 6) MINOR

Filed 7) AVGBAL

Field 8) SERVICECHARGETYPE

Filed 9) SUFFIX

Field 10) ACCOUNTTYPE

Field 11) INTERESTRATE

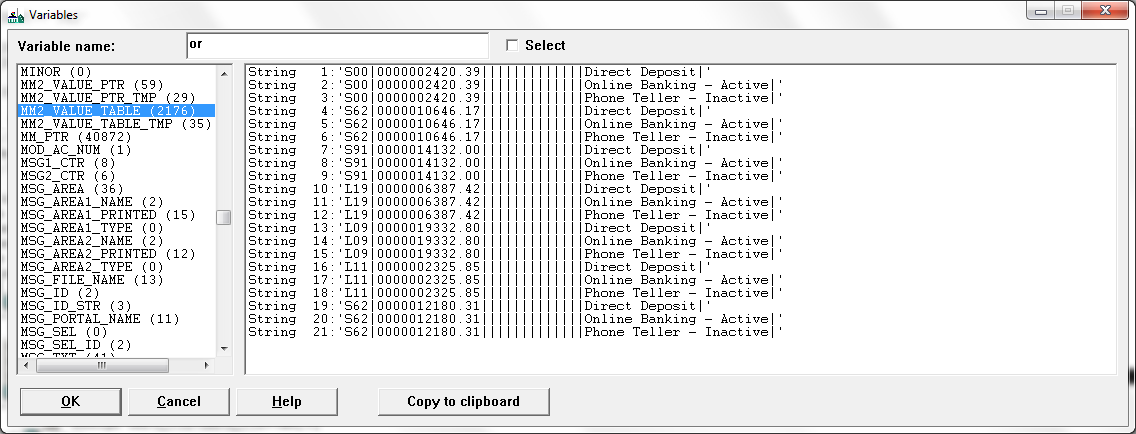
Field 12) SHAREDESC

Field 13) LOANDESC

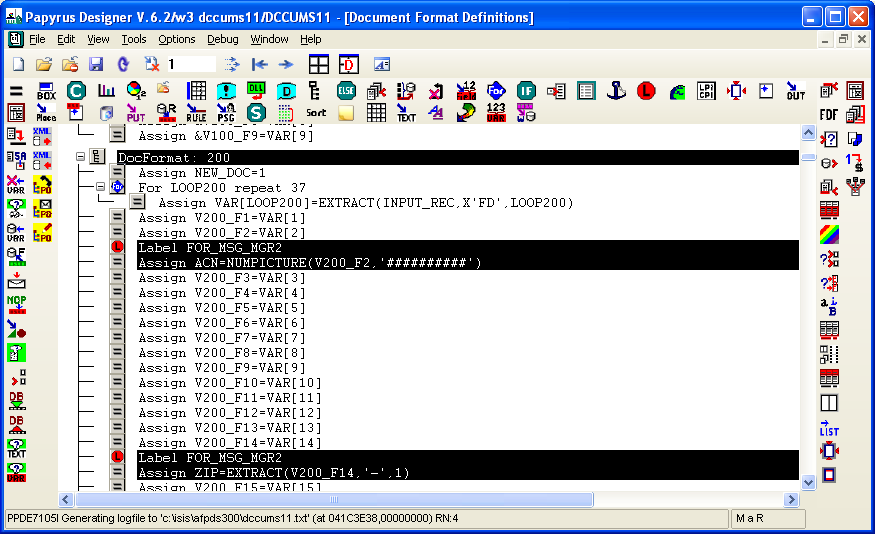
Field 14) SHAREID

Field 15) ESERVICE

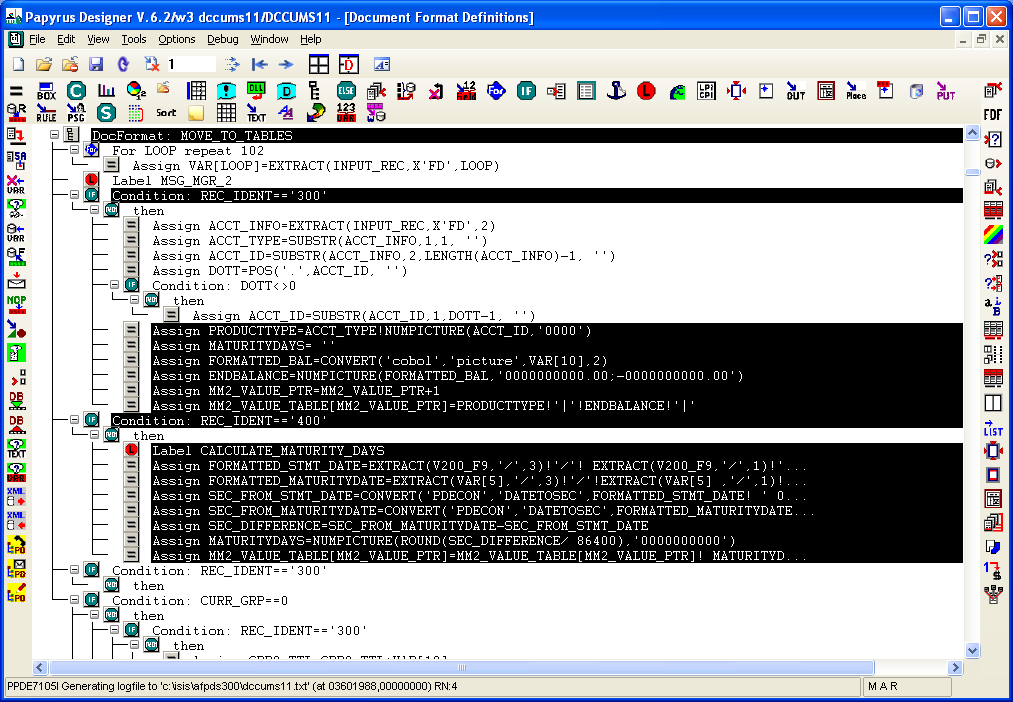
This is an example of how the MM2\_VALUE\_TABLE array looks like.



* ACN is important for setting selective messaging by account number. Therefore, make sure that variable has a value assigned to it. This screenshot below is applicable to ULTRA data core ONLY. Please do NOT copy this logic if you are NOT setting up for a ULTRA data application.
* ZIP is required for setting selective messaging by zip code. Therefore, make sure to assign a value for that variable. Once again, this below example ONLY applies to ULTRA data. Please make sure ZIP holds 5 digits.
* Since account number and zip code only hold 1 single value, it is NOT necessary to store them in a table like in the next screenshot.

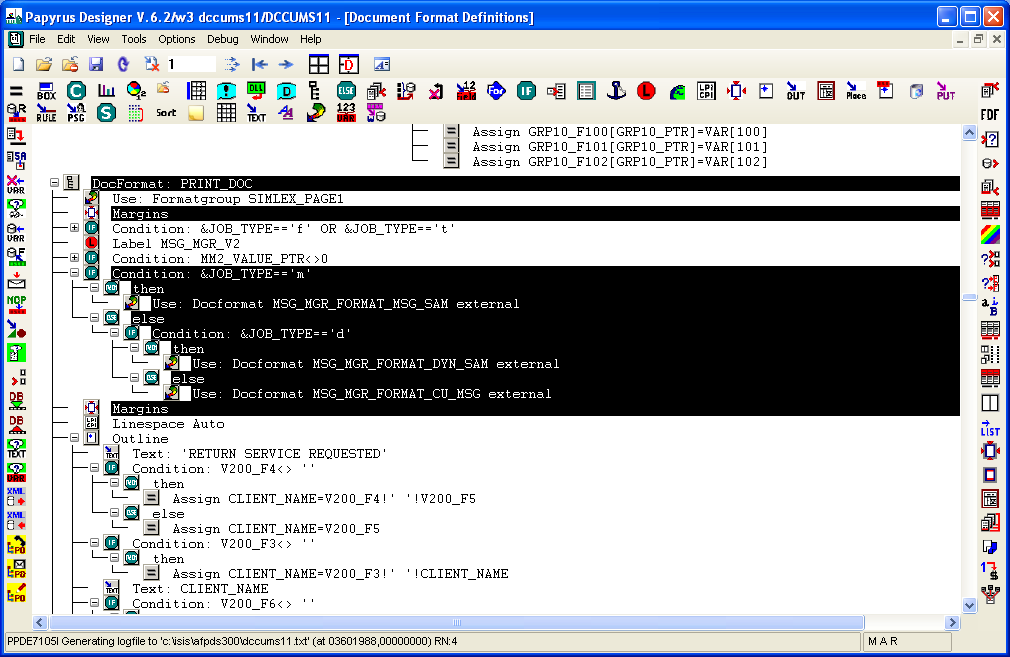


* This screenshot below illustrates selective criteria’s value capturing for ULTRA data. This is the most essential step of message manager 2.2 setup. Depending on the type of data core, the values are captured differently. Please be sure to pull values from the correct data fields. If the values stored in this step are wrong, the message to be displayed will also be wrong.
* This part should be done prior to displaying the message, which most likely will be handled at PRINT\_DOC docformat
* As mentioned above, NOT all selective criteria need to be stored in a table. This only applies to selective criteria that hold multiple values such as PRODUCTTYPE, since a statement might contain different types of PRODUCTs like Checking, Savings, CD, IRA, etc. For a visa application, which only contains a visa product, you do NOT need a table to store the criteria. You can just follow previous sample on how to set up selective messaging by account number and zip code.

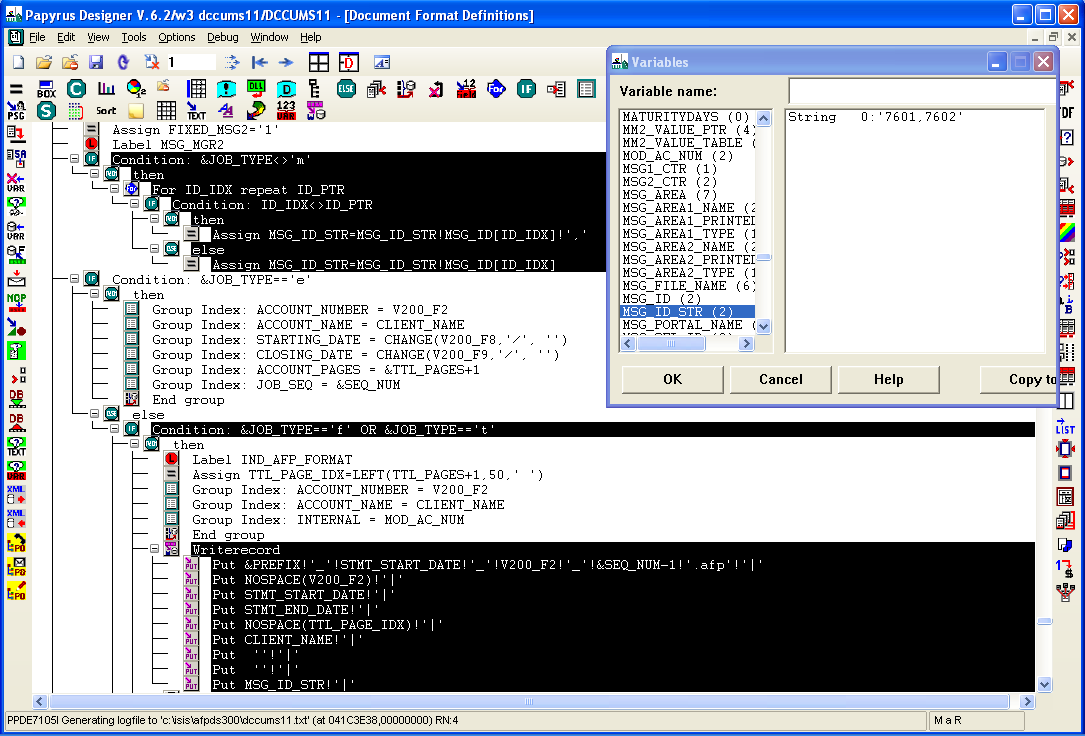


Step 4: Display the targeted message on the statement. Don’t forget to test out all selective criteria thoroughly before proceeding.

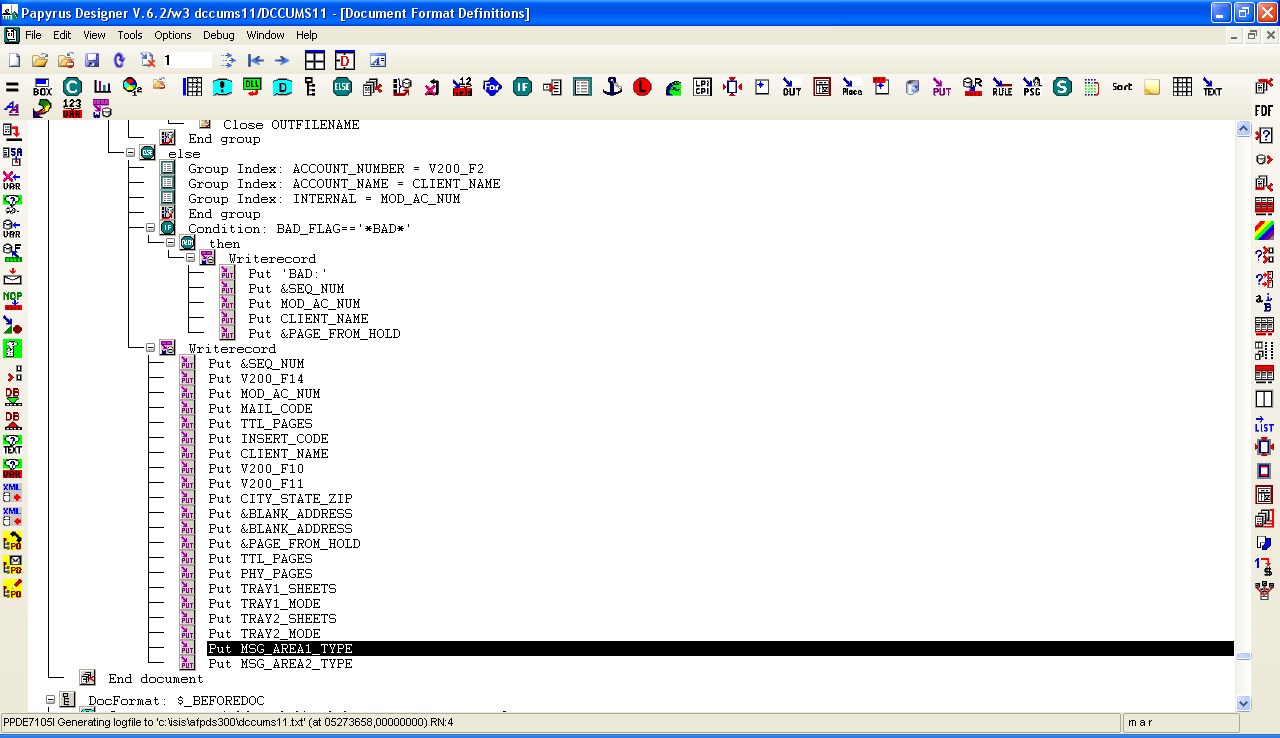
* NOTE: This part of coding is to be added at the beginning of PRINT\_DOC Docformat AFTER the criteria evaluation code. Be sure to reset the margin to its original values after the message was printed. Otherwise, the display of the statement content will be off.

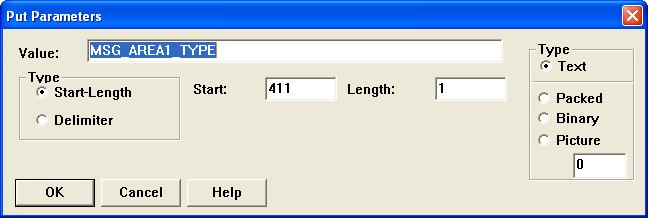


Step 5: Update e-statement index file to have the new format (pipe delimited), where the message id string is required to be added to field 9th.



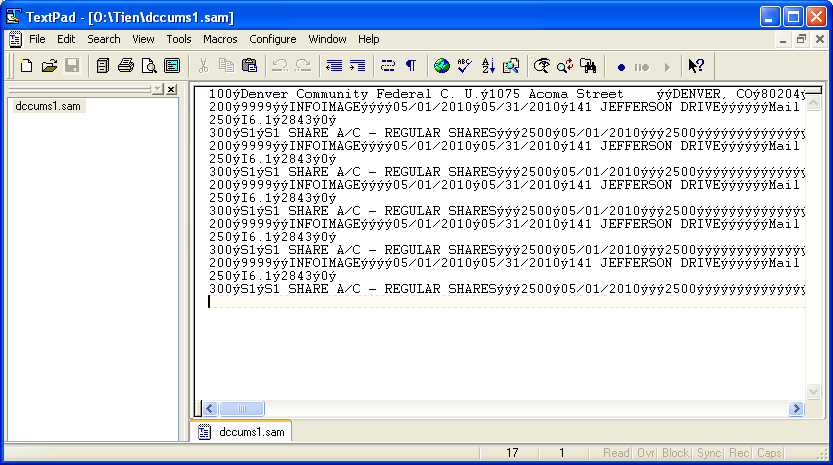
Step 6: Output the type of message box #1 (0 – default, 1 – type 1, etc.) to position 411 of the txt record. For message box #2’s type, use position 412.

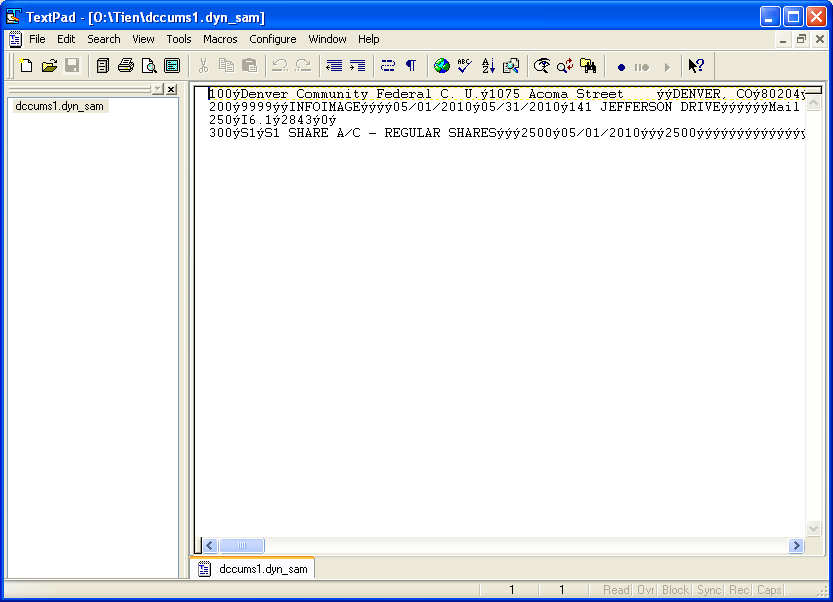


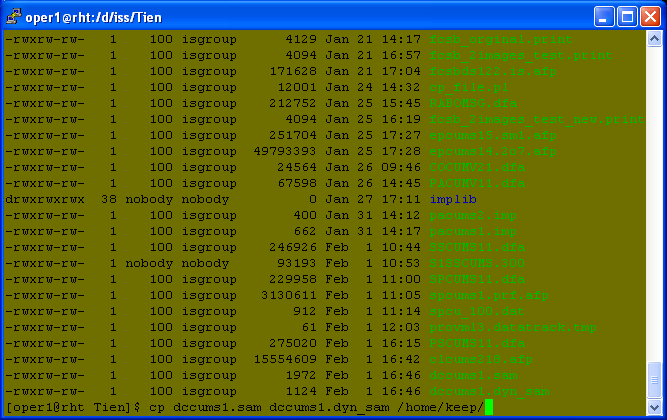


Step 7: Create sample files $prefix.sam (with at least 5 accounts) and $prefix.dyn\_sam (with just 1 account), and copy them to /home/keep/

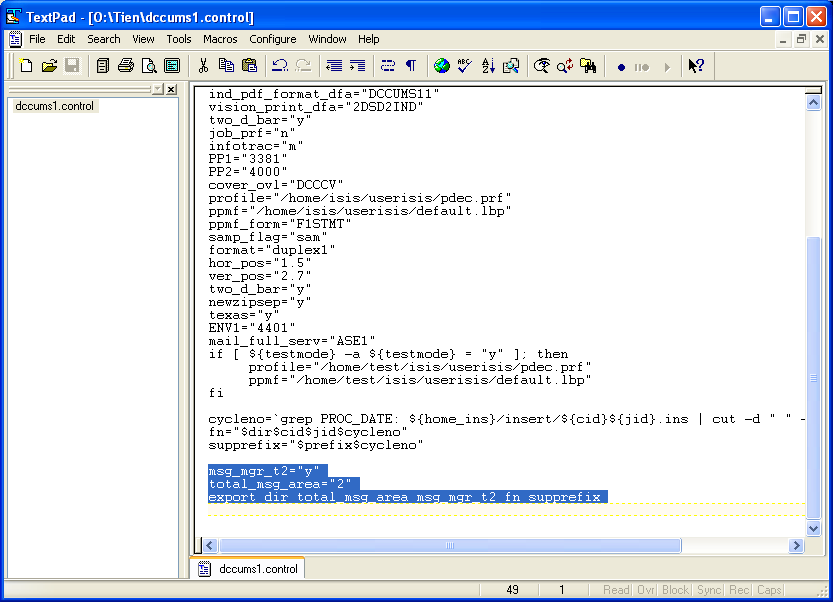
NOTE: You can reuse sample files from the same core. For example, if you are setting up message manager for ULTRA data, you can copy /home/keep/dccums1.sam and /home/keep/dccums1.dyn\_sam, and name them to the corresponding files using your prefix. This is because DCCU’s statement file is of ULTRA data.







Step 8: Update the control file to add the followings: msg\_mgr\_t2=”y”; total\_msg\_area=”total number of messages”; export msg\_mgr\_t2 total\_msg\_area;



*FYI:* ***If*** *client wants color message(s) in PDFs, add the color\_flag in control file and export. Estatement*

*uses the tiff image client entered, paper option cannot be color, because the printer cannot print color. Therefore, the color\_flag is for PDF options only (pdfi, pdf1, or e) to convert the color TIFF to a color PSEG. Set flag for whichever option is needed only.*

*For example: UTCU uses pdfi option for PDFs. In control file, add* **color\_flag=“pdfi”** *and export*

Step 9: Move the updated format dfa, and control file to rht.

NOTE: Remember to move all resources (ovl, tiff, pseg) to preprocessing02 /home/afp2web/isis/… folders. Also, please change the extension of TIF to tif (lowercase). Otherwise, you will see missing resources on the sample statement.

Step 10: Inform ISD to create a sample cycle. If there are errors, ask them to provide the command which generates the errors. Here is an example:

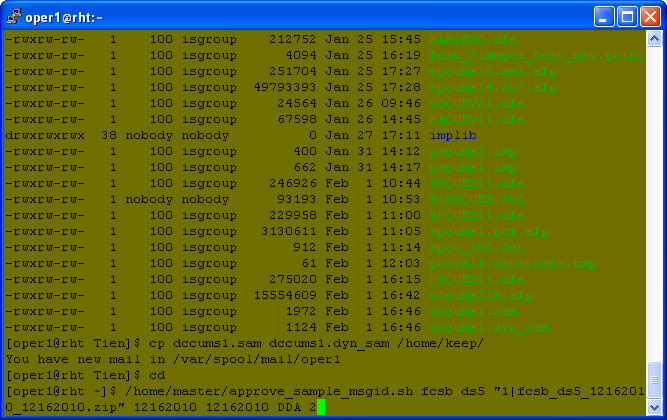
Send IS command **/home/master/approve\_sample\_msgid.sh fcsb ds5 "1|fcsb\_ds5\_12162010\_12162010.zip" 12162010 12162010 DDA 2** to **10.8.9.68**:3490

Reply from IS: [4|unable to transfer pdf file ]

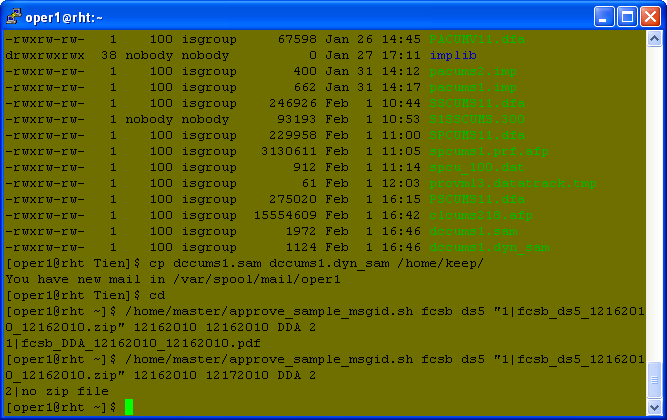
Then, test out the command on the server ISD tried to send it to.

10.8.9.68 is RHT

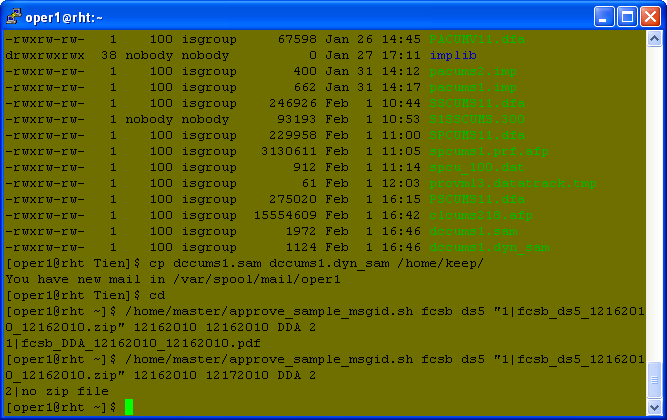
10.8.9.61 is RHS



Sample of a successful command: [return code 1|pdf file name]



Sample of a failed command: [return error code #|error message]



Debug and fix the error. Then, inform ISD to resume the test.

Step 11: Migrate updated programs to RHS. To query ISD’s staging system, set job\_id = “99999” or “999999”. To access their pilot system, set job\_id = “88888” or “888888”.

ISD’s pilot URL: [http://10.8.8.213:9090/MsgServlet22?cid=${cid}&apptype=${app}&cycledate=${cycledate](http://10.8.8.213:9090/MsgServlet22?cid=$%7bcid%7d&apptype=$%7bapp%7d&cycledate=$%7bcycledate)}

ISD’s staging URL: [http://10.8.8.214:9090/MsgServlet22?cid=${cid}&apptype=${app}&cycledate=${cycledate](http://10.8.8.214:9090/MsgServlet22?cid=$%7bcid%7d&apptype=$%7bapp%7d&cycledate=$%7bcycledate)}

ISD’s production URL:<http://10.8.9.20:9090/MsgServlet22?cid=${cid}&apptype=${app}&cycledate=${cycledate>}

Step 12: Process a big data file in the P/R environment to test for selective criteria after a sample message has been created by ISD. The standard program will base on the date defined in PROC\_DATE of the insert file to pull the corresponding message when you process the job. So, make sure a particular test message for the date specified in PROC\_DATE has already been created for you. Otherwise, your process will fail. When this happens, ask ISD to either back date a test message that was already created or create a new one for you.

**Part II:** **Set up *free-flow* message**

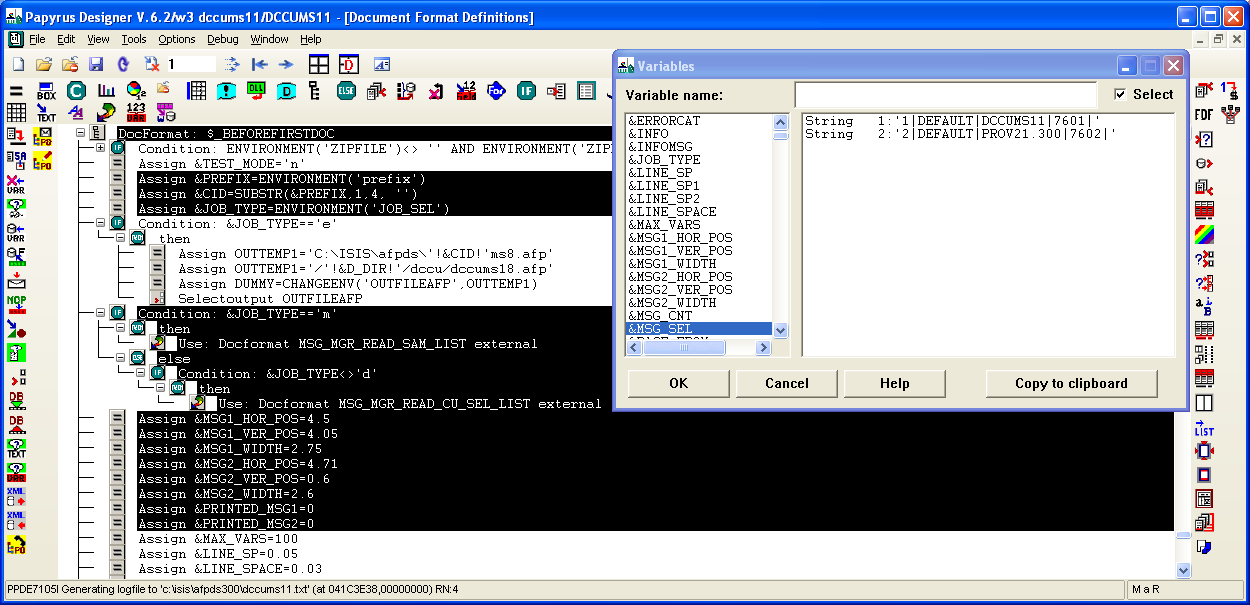
Step 0: Provide ISD the message information such as: width, height, font name, font size, maximum number of lines, maximum characters, etc.

* NOTE: Here are some model programs (by data core) you can use as a reference. Pick the one that most reflects your data core.

1. PRINT IMAGE - OSCUMS11.dfa, OSCUMV11.dfa, NTSBLS11.dfa
2. OSI – PROVMS11.dfa, SVBTMS11.dfa
3. ULTRA DATA – DCCUMS11.dfa
4. SYMITAR DATA – NMCUMS11.dfa
5. MISER – GECUMS11.dfa

Step 1: Set up global variables such as the horizontal and vertical positions of the message, and store selective criteria entries from the index file ($prefix\_msg\_portal.txt) in &MSG\_SEL array.

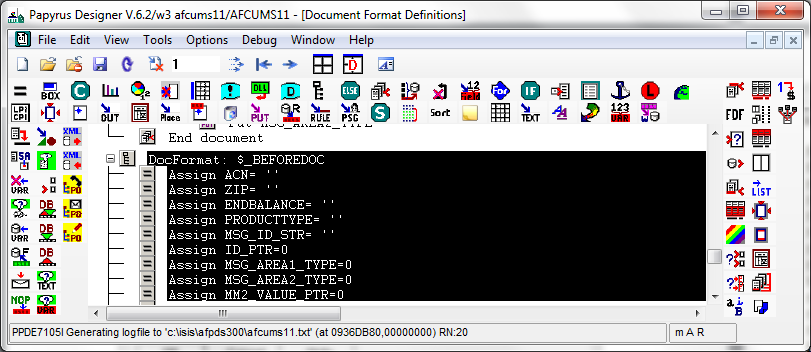
* NOTE: The coding below should be added to BEFOREFIRSTDOC Docformat.



* /home/keep/msgmgr2\_sample\_portal.txt is an example of how an index file with selective criteria looks like. You can rename this file using the specified naming convention ($prefix\_msg\_portal.txt), and modify it accordingly for testing purpose.

Step 2: Initialize local variables such as selective criteria name (e.g. ACN = selective by account number, ZIP = selective by zip code, etc.)

* NOTE: The coding below should be added to BEFOREDOC Docformat.



NOTE: \*\*\*SKIP THIS STEP 3 IF YOU ARE SETTING UP MESSAGE MANAGER 1.0 – NO SELECTIVE MESSAGING\*\*\*\*  
Step 3: Extract/Calculate and store selective criteria information in a predefined table (see below for the table’s definition) called MM2\_VALUE\_TABLE indexed by MM2\_VALUE\_PTR **(Please be sure to use these names)**. Table is only needed if the criteria have more than 1 possible value such as product type. For example, a product type can be of Checking, Savings, CD, or Money Market, etc. If ALL your selective criteria have single value, please do the following:

1. Assign MM2\_VALUE\_PTR = 1 (done at BEFOREDOC docformat)
2. Assign MM2\_VALUE\_TABLE[1] = ‘’ (empty) (done at BEFOREDOC docformat)

* Definition for MM2\_VALUE\_TABLE: (It is pipe ‘|’ delimited)

**\*\*IMPORTANT\*\***: If there is any selective criteria that CANNOT be found in the list below, please let Tien know to add it to the table.

Field 1) PRODUCTTYPE

Field 2) ENDBALANCE/ENDINGBAL

Field 3) MATURITYDAYS

Field 4) APPLICATIONCODE

Field 5) MAJOR

Field 6) MINOR

Filed 7) AVGBAL

Field 8) SERVICECHARGETYPE

Filed 9) SUFFIX

Field 10) ACCOUNTTYPE

Field 11) INTERESTRATE

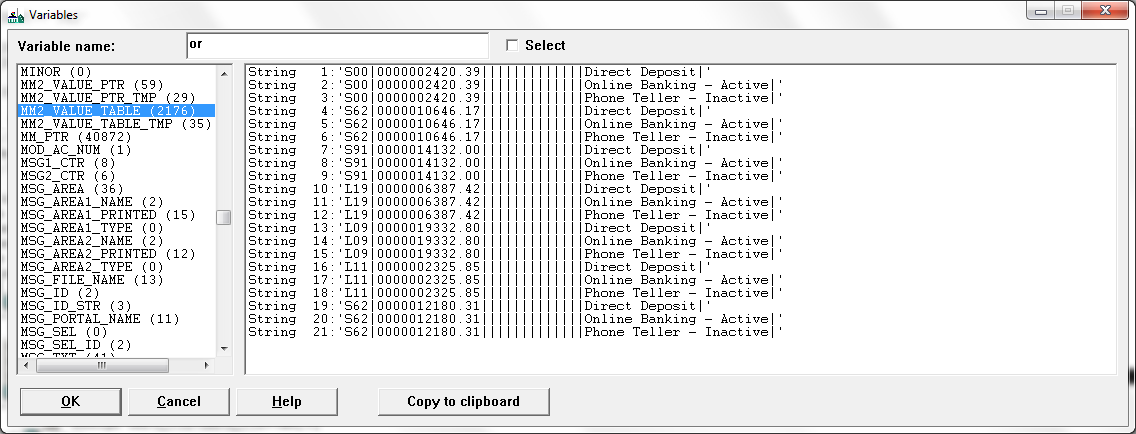
Field 12) SHAREDESC

Field 13) LOANDESC

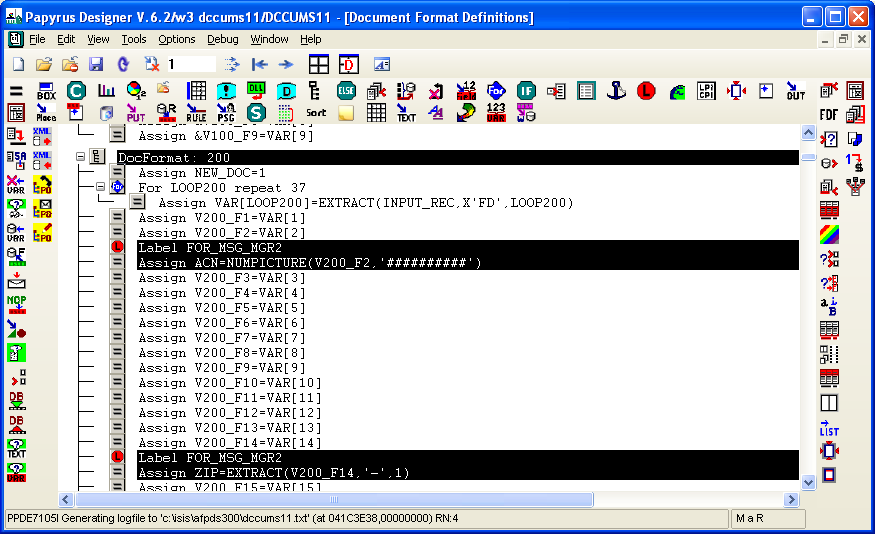
Field 14) SHAREID

Field 15) ESERVICE

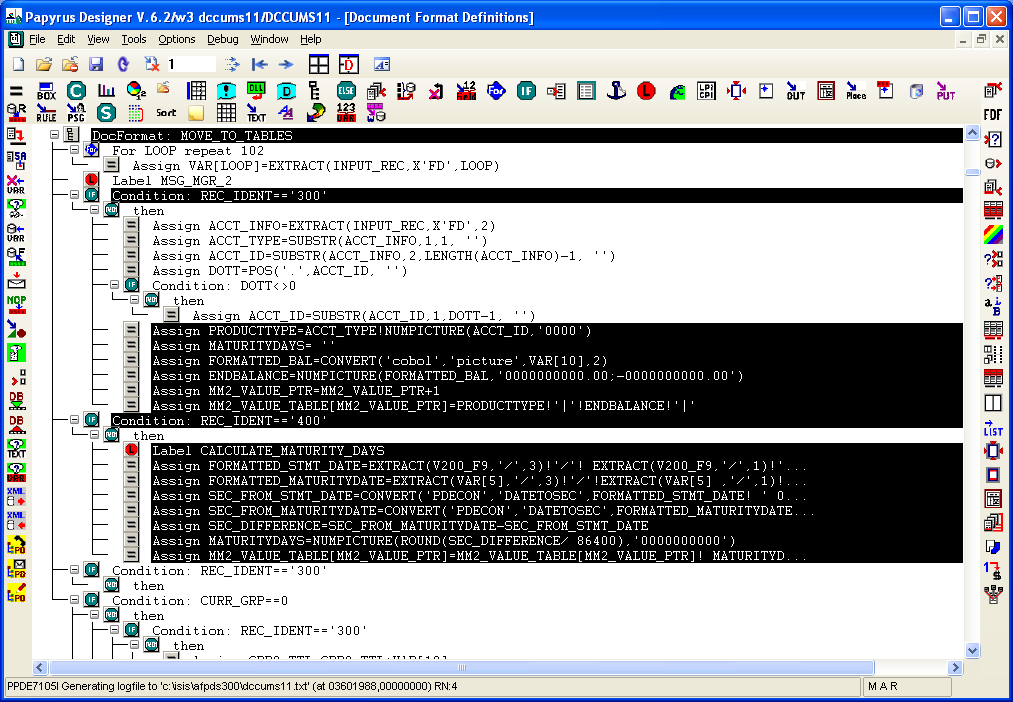
This is an example of how the MM2\_VALUE\_TABLE array looks like.



* ACN is important for setting selective messaging by account number. Therefore, make sure that variable has a value assigned to it. This screenshot below is applicable to ULTRA data core ONLY. Please do NOT copy this logic if you are NOT setting up for a ULTRA data application.
* ZIP is required for setting selective messaging by zip code. Therefore, make sure to assign a value for that variable. Once again, this below example ONLY applies to ULTRA data.
* Since account number and zip code only hold 1 single value, it is NOT necessary to store them in a table like in the next screenshot.

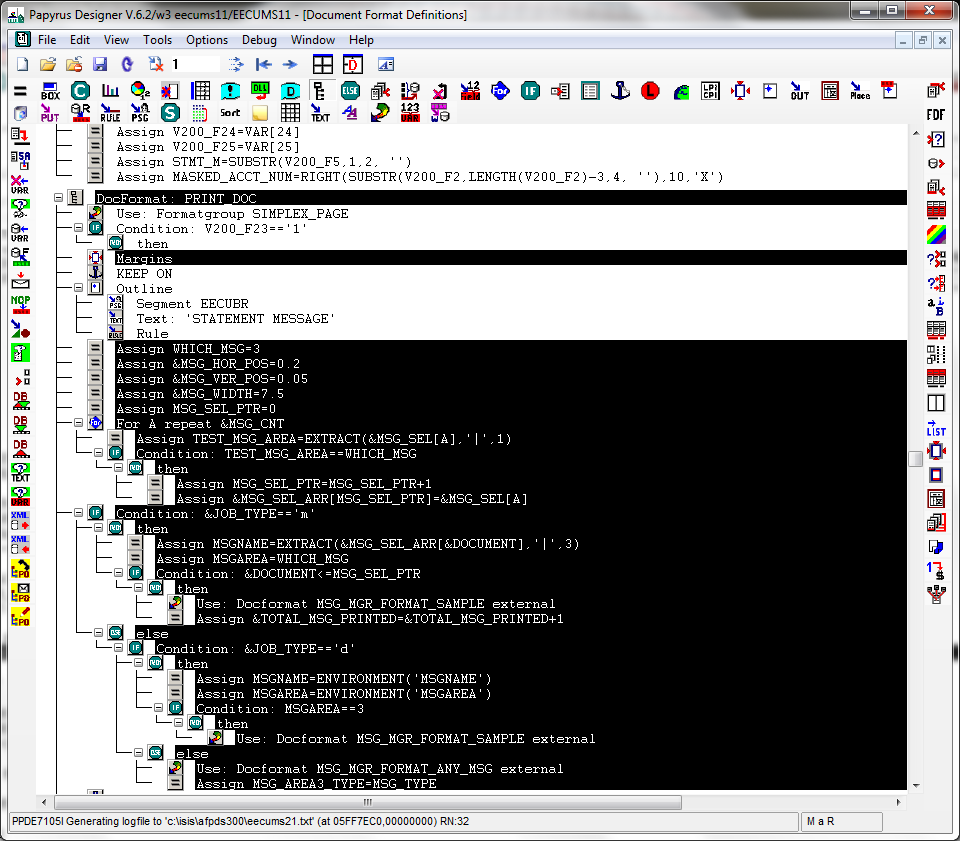


* This screenshot below illustrates selective criteria’s value capturing for ULTRA data. This is the most essential step of message manager 2.2 setup. Depending on the type of data core, the values are captured differently. Please be sure to pull values from the correct data fields. If the values stored in this step are wrong, the message to be displayed will also be wrong.
* This part should be done prior to displaying the message, which most likely will be handled at PRINT\_DOC docformat
* As mentioned above, NOT all selective criteria need to be stored in a table. This only applies to selective criteria that hold multiple values such as PRODUCTTYPE, since a statement might contain different types of PRODUCTs like Checking, Savings, CD, IRA, etc. For a visa application, which only contains a visa product, you do NOT need a table to store the criteria. You can just follow previous sample on how to set up selective messaging by account number and zip code.

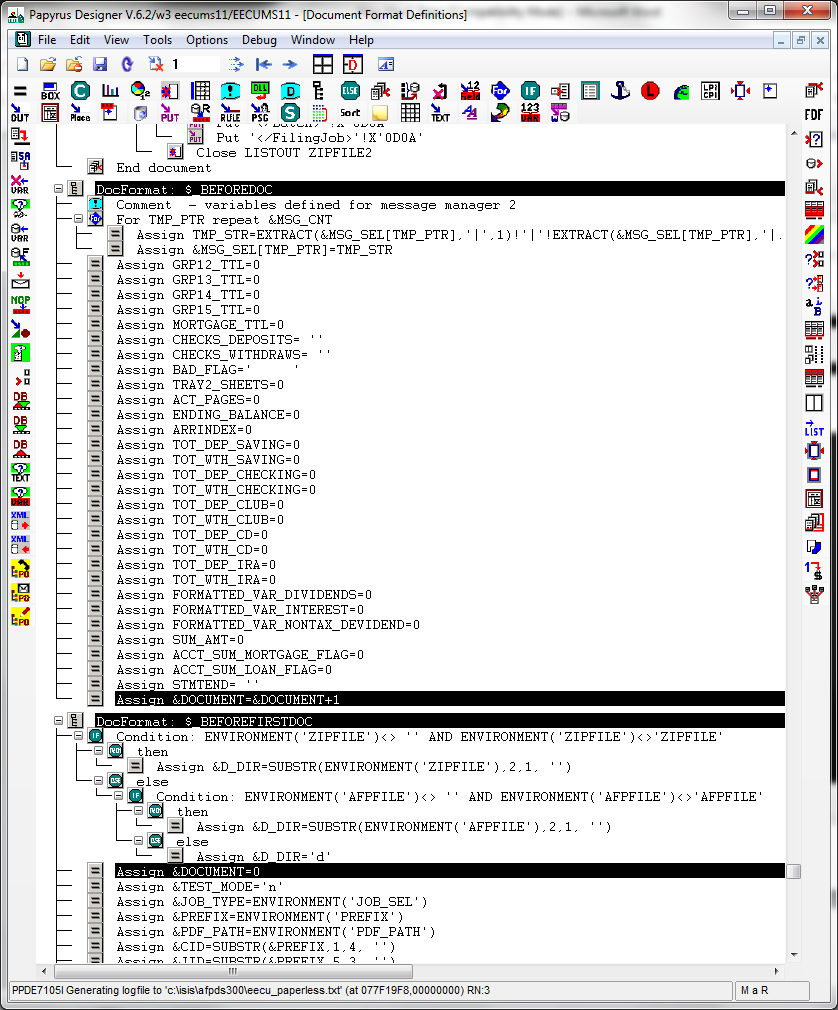


Step 4: Display the targeted message on the statement. Don’t forget to test out all selective criteria thoroughly before proceeding. (Use *EECUMS11.dfa* as a sample)

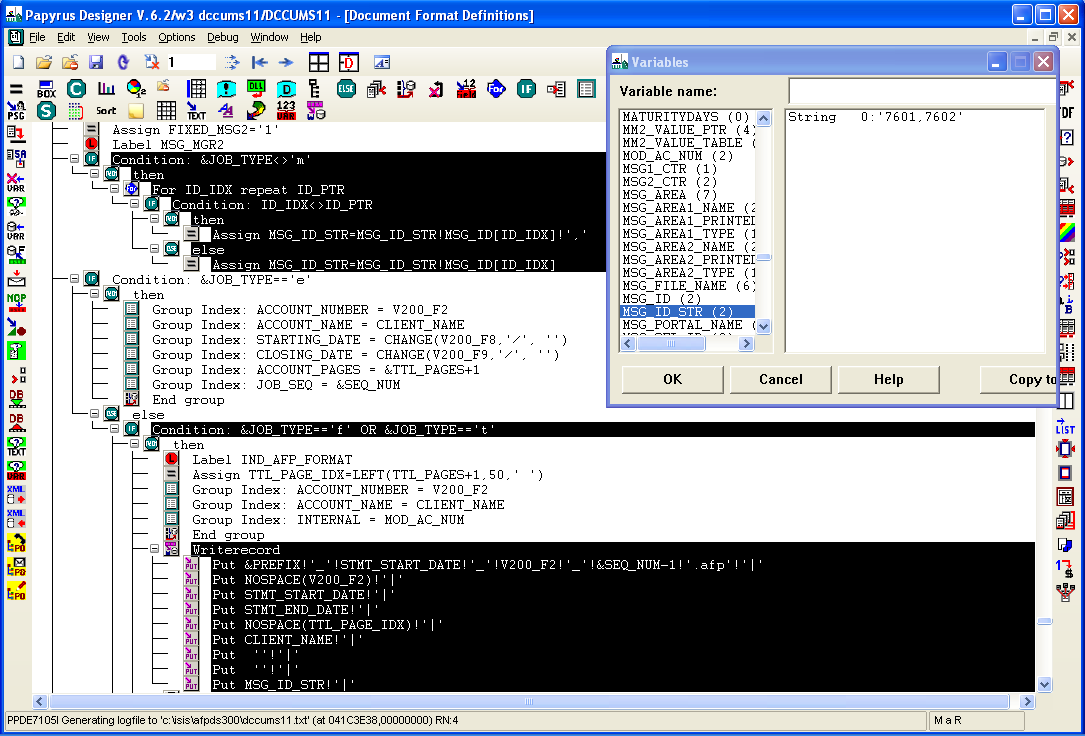
* NOTE: This part of coding is to be added to PRINT\_DOC Docformat AFTER the criteria evaluation code. Be sure to reset the margin to its original values after the message was printed. Otherwise, the display of the statement content will be off.



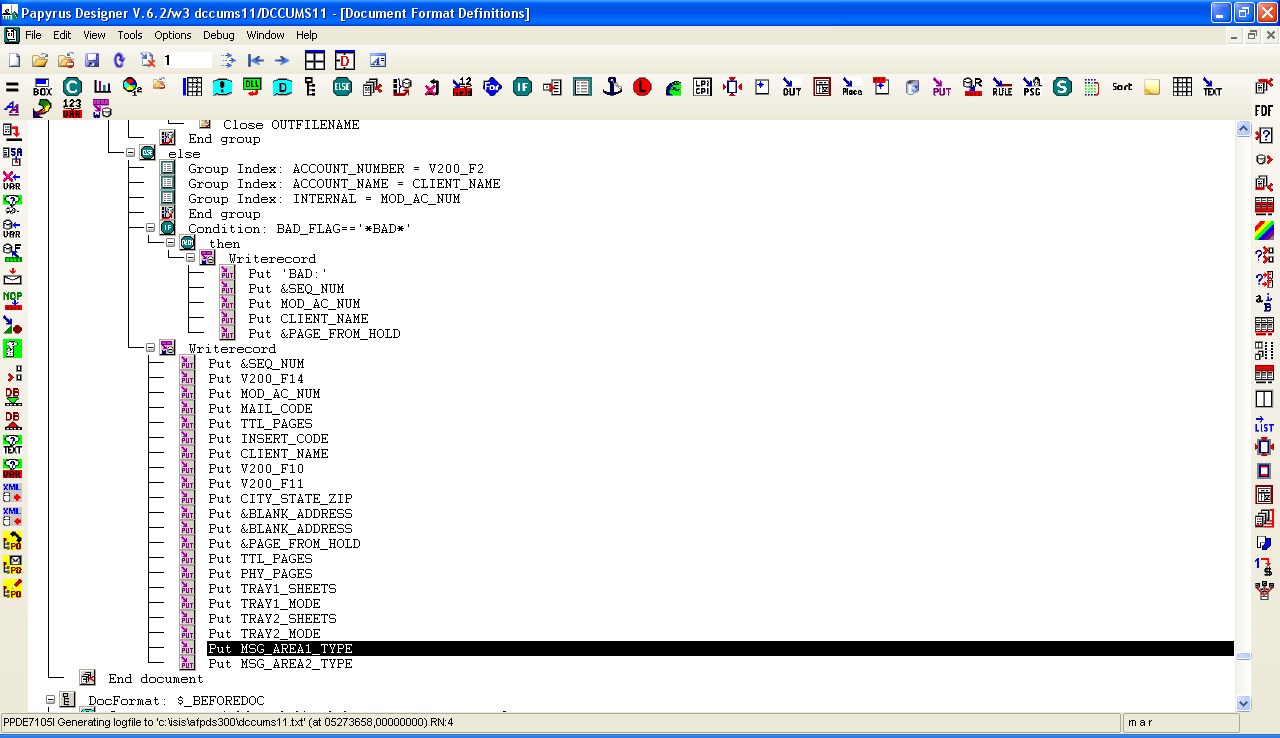
* This is to be added to BEFOREDOC and BEFOREFIRSTDOC Docformats.

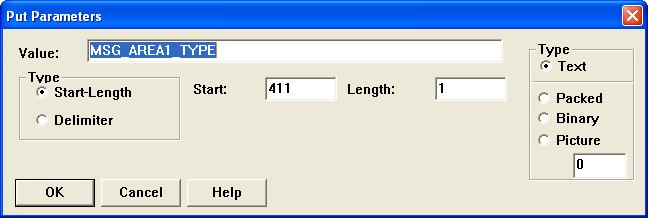


Step 5: Update e-statement index file to have the new format (pipe delimited), where the message id string is required to be added to field 9th.



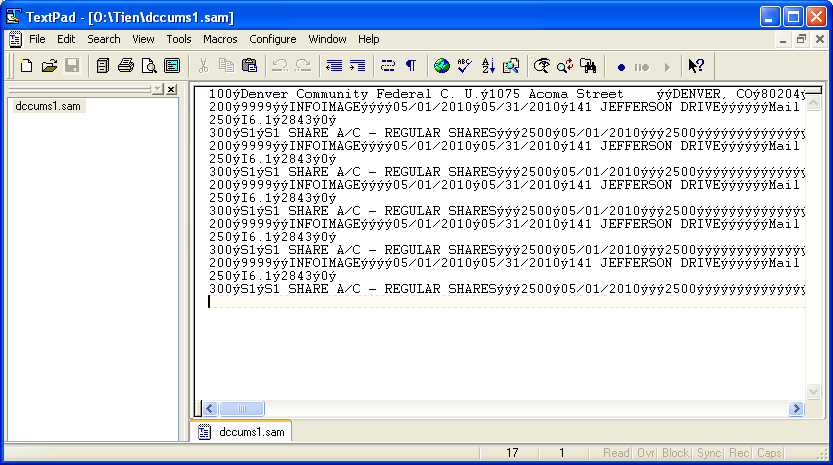
Step 6­: Output the type of message box #1 (0 – default, 1 – type 1, etc.) to position 411 of the txt record. For message box #2’s type, use position 412.

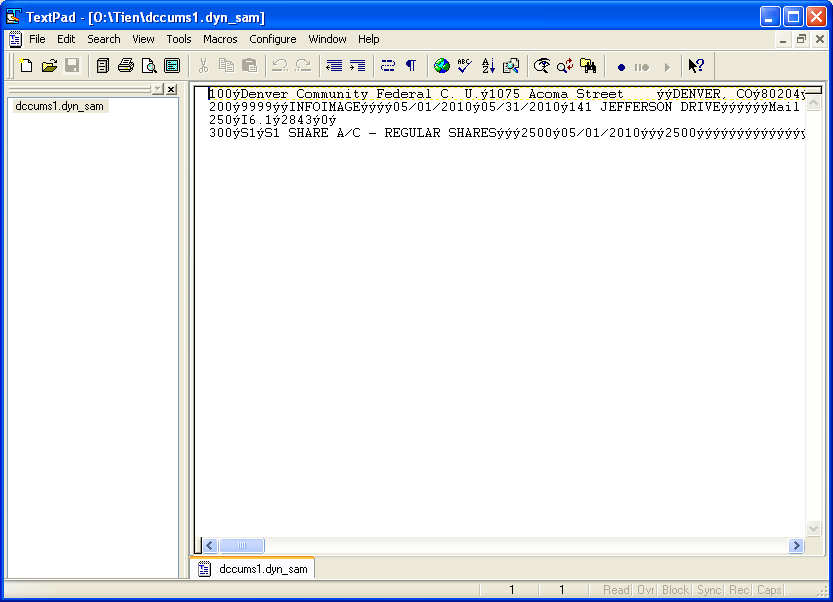


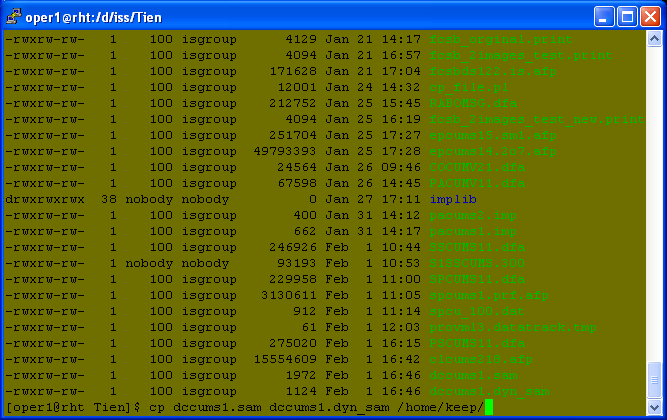


Step 7: Create sample files $prefix.sam (with at least 5 accounts) and $prefix.dyn\_sam (with just 1 account), and copy them to /home/keep/

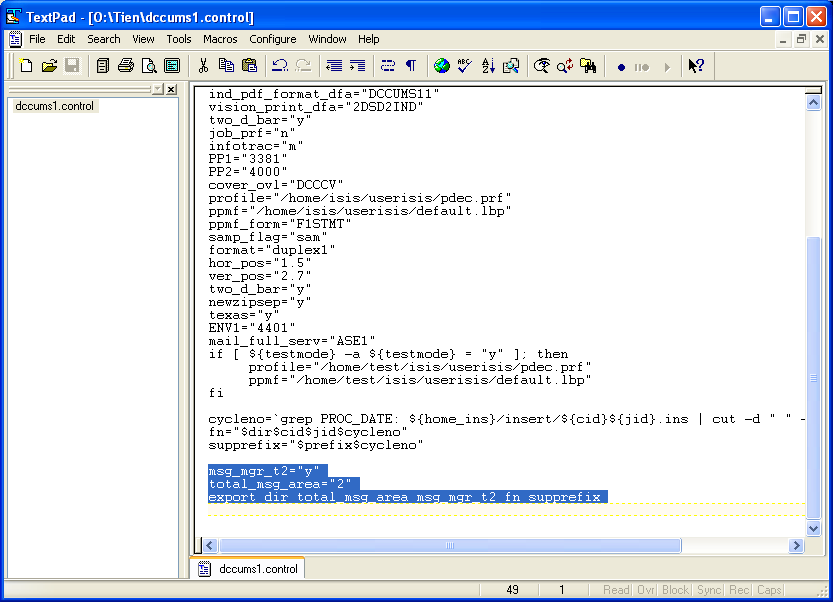
NOTE: You can reuse sample files from the same core. For example, if you are setting up message manager for ULTRA data, you can copy /home/keep/dccums1.sam and /home/keep/dccums1.dyn\_sam, and name them to the corresponding files using your prefix. This is because DCCU’s statement file is of ULTRA data.







Step 8: Update the control file to add the followings: msg\_mgr\_t2=”y”; total\_msg\_area=”total number of messages”; export msg\_mgr\_t2 total\_msg\_area;



*FYI:* ***If*** *client wants color message(s) in PDFs, add the color\_flag in control file and export. Estatement*

*uses the tiff image client entered, paper option cannot be color, because the printer cannot print color. Therefore, the color\_flag is for PDF options only (pdfi, pdf1, or e) to convert the color TIFF to a color PSEG. Set flag for whichever option is needed only.*

*For example: UTCU uses pdfi option for PDFs. In control file, add* **color\_flag=“pdfi”** *and export*

Step 9: Move the updated format dfa, and control file to rht.

NOTE: Remember to move all resources (ovl, tiff, pseg) to preprocessing02 /home/afp2web/isis/… folders. Also, please change the extension of TIF to tif (lowercase). Otherwise, you will see missing resources on the sample statement.

Step 10: Inform ISD to create a sample cycle. If there are errors, ask them to provide the command which generates the errors. Here is an example:

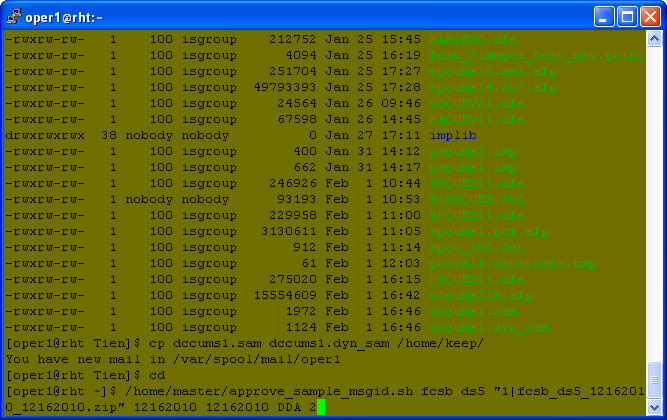
Send IS command **/home/master/approve\_sample\_msgid.sh fcsb ds5 "1|fcsb\_ds5\_12162010\_12162010.zip" 12162010 12162010 DDA 2** to **10.8.9.68**:3490

Reply from IS: [4|unable to transfer pdf file ]

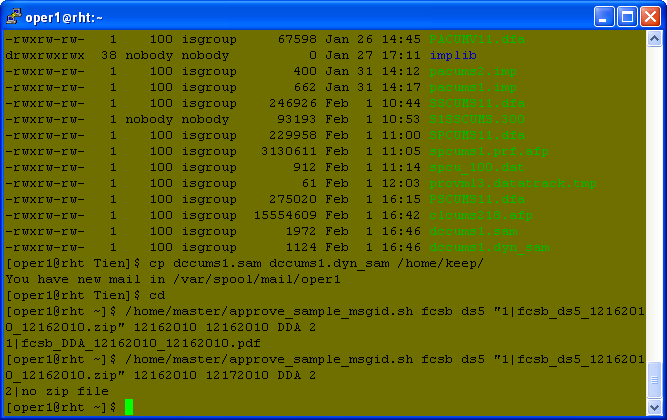
Then, test out the command on the server ISD tried to send it to.

10.8.9.68 is RHT

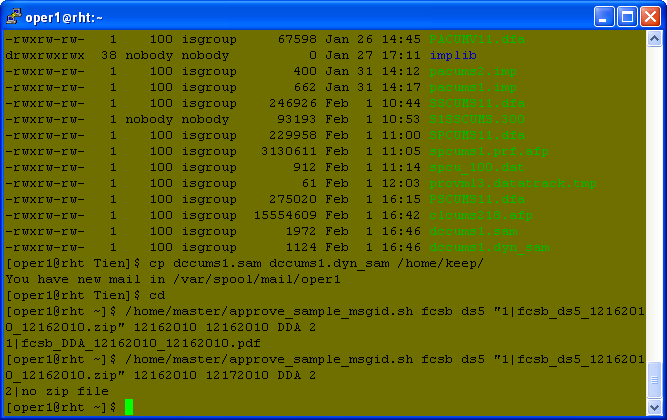
10.8.9.61 is RHS



Sample of a successful command: [return code 1|pdf file name]



Sample of a failed command: [return error code #|error message]



Debug and fix the error. Then, inform ISD to resume the test.

Step 11: Migrate updated programs to RHS. To query ISD’s staging system, set job\_id = “99999” or “999999”. To access their pilot system, set job\_id = “88888” or “888888”.

ISD’s pilot URL: [http://10.8.8.213:9090/MsgServlet22?cid=${cid}&apptype=${app}&cycledate=${cycledate](http://10.8.8.213:9090/MsgServlet22?cid=$%7bcid%7d&apptype=$%7bapp%7d&cycledate=$%7bcycledate)}

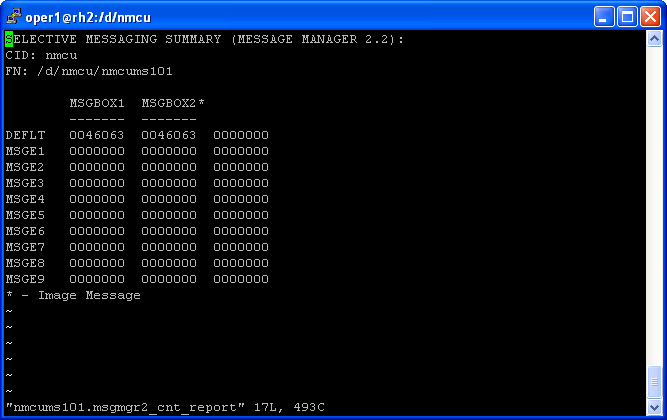
ISD’s staging URL: [http://10.8.8.214:9090/MsgServlet22?cid=${cid}&apptype=${app}&cycledate=${cycledate](http://10.8.8.214:9090/MsgServlet22?cid=$%7bcid%7d&apptype=$%7bapp%7d&cycledate=$%7bcycledate)}

ISD’s production URL:<http://10.8.9.20:9090/MsgServlet22?cid=${cid}&apptype=${app}&cycledate=${cycledate>}

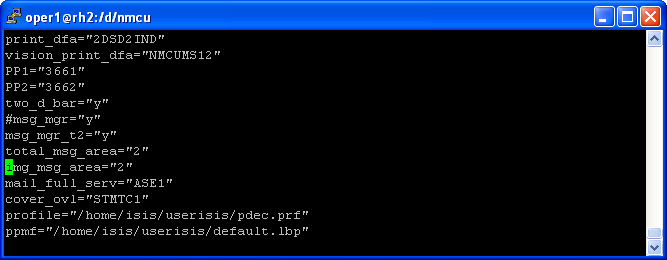
Step 12: Process a big data file in the P/R environment to test for selective criteria after a sample message has been created by ISD. The standard program will base on the date defined in PROC\_DATE of the insert file to pull the corresponding message when you process the job. So, make sure a particular test message for the date specified in PROC\_DATE has already been created for you. Otherwise, your process will fail. When this happens, ask ISD to either back date a test message that was already created or create a new one for you.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Utility Programs and Special Notes\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. For online proofing, the afp proof file will be converted to pdf and sent to ISD.
2. Property of a grayscale pseg image: Image Format: IOCA, Function Set: FS45 – JPEG, Original Color = CMYK, Image Compression: JPEG, Number of Colors: 256
3. Message Count Report: ${fn}.msgmgr2\_cnt\_report (generated by /home/master/msg\_mgr2\_count\_report.pl)

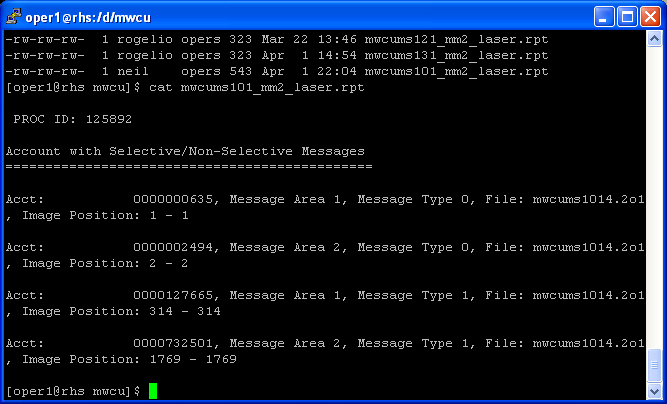


- The \* next to the MSGBOX2 on the above screenshot indicates that the message area 2 is a tiff image. This is controlled by a variable called img\_msg\_area. The image control variable is set in the control file as followed:



1. QA file for LASER department:

${prefix}\_mm2\_laser.rpt (Generated by /home/master/msg\_mgr2\_get\_selmsg.pl)



1. Email attachment of Pdf sample for QA: (Generated by /home/master/msg\_mgr2\_convert\_sample.sh)

Create a distribution list called mail\_list\_msgmgr2=Email\_Address1,Email\_Address2, etc in /home/keep/maillist/${cid}\_mail.lis

