Paper 1384-2015

Building Flexible Jobs in DataFlux®

Jeanne Estridge, Sinclair Community College

ABSTRACT

Creating DataFlux® jobs that can be executed from job scheduling software can be challenging. This paper describes how to create a process job that calls an email node that accepts a distribution list, subject line, email text, and attachment file name as macro variables. It also explains how to execute this job from a command line, overriding those values at run-time.

INTRODUCTION

Creating flexible batch jobs in DataFlux® requires the construction of two different kinds DataFlux® jobs: data jobs, which execute tasks like diagnosing data quality issues or sending emails, and process jobs, which allow grouping of related data jobs and support logical decisions, looping and other features that are not available in data jobs. Since process jobs can be called from the command line, they can be scheduled with whatever job scheduling software your site uses.

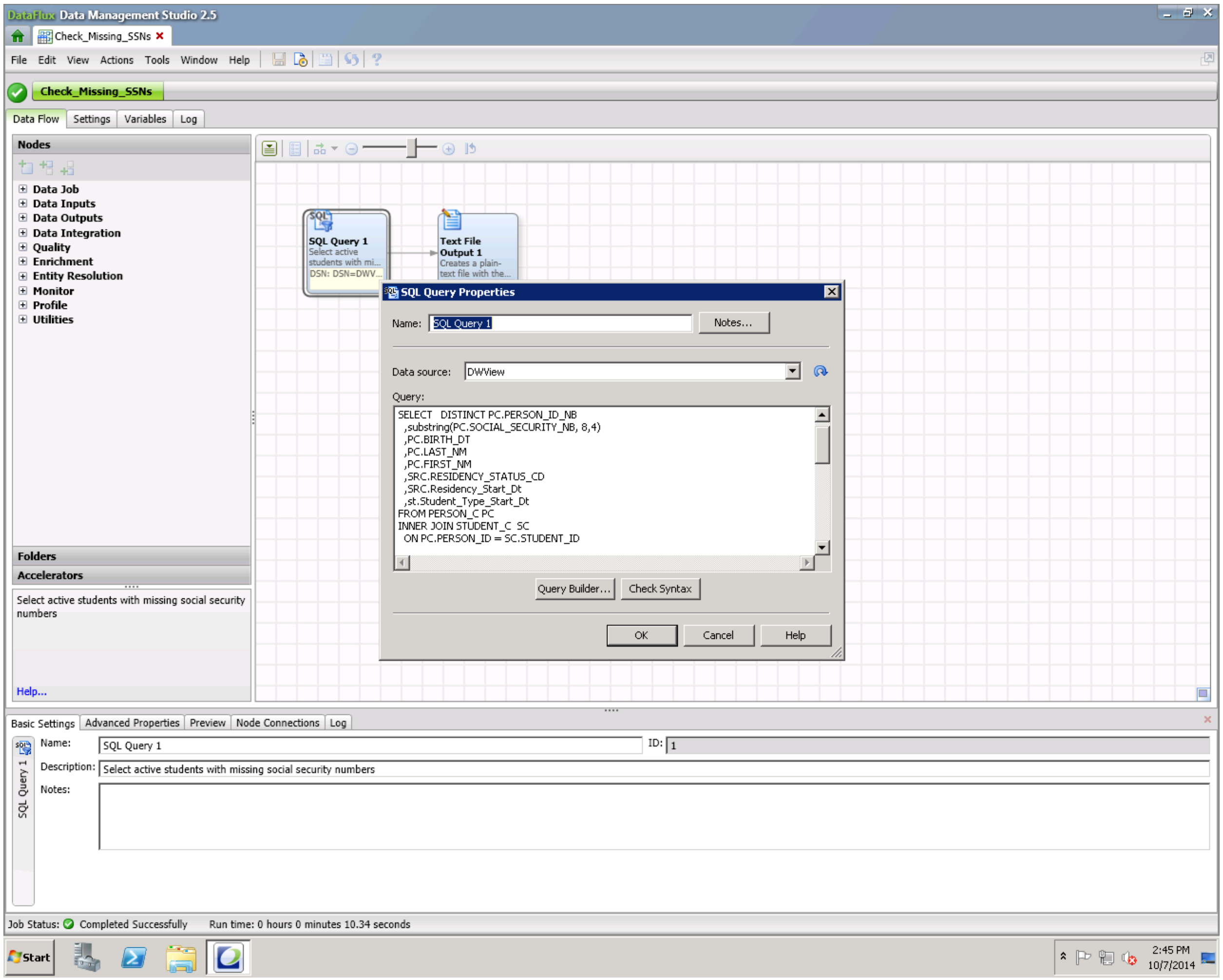
Once you've created these jobs, you'll create global macro variables and use them to replace the parameters in the email job. Then you'll create local macro variables and associate them to the global variables, giving you flexibility. Finally, you'll create a command to execute the process job.

CREATING THE DATA JOBS

You will have two separate data jobs:

1) The first will have an input node that extracts the information you wish to report, along with an output node that writes a text output file of that information to disk.
2) The second is an email node that sends the text output file to a distribution list.

Figure 1. Create the Data Query



Select active students with missing social security numbers

Name: SQL Query 1

Data source: DWView

Query:

```
SELECT   DISTINCT PC.PERSON_ID_NB
  ,substring(PC.SOCIAL_SECURITY_NB, 8,4)
  ,PC.BIRTH_DT
  ,PC.LAST_NM
  ,PC.FIRST_NM
  ,SRC.RESIDENCY_STATUS_CD
  ,SRC.Residency_Start_Dt
  ,st.Student_Type_Start_Dt
FROM PERSON_C PC
INNER JOIN STUDENT_C  SC
  ON PC.PERSON_ID = SC.STUDENT_ID
```

Name: SQL Query 1 ID: 1

Description: Select active students with missing social security numbers

Notes:

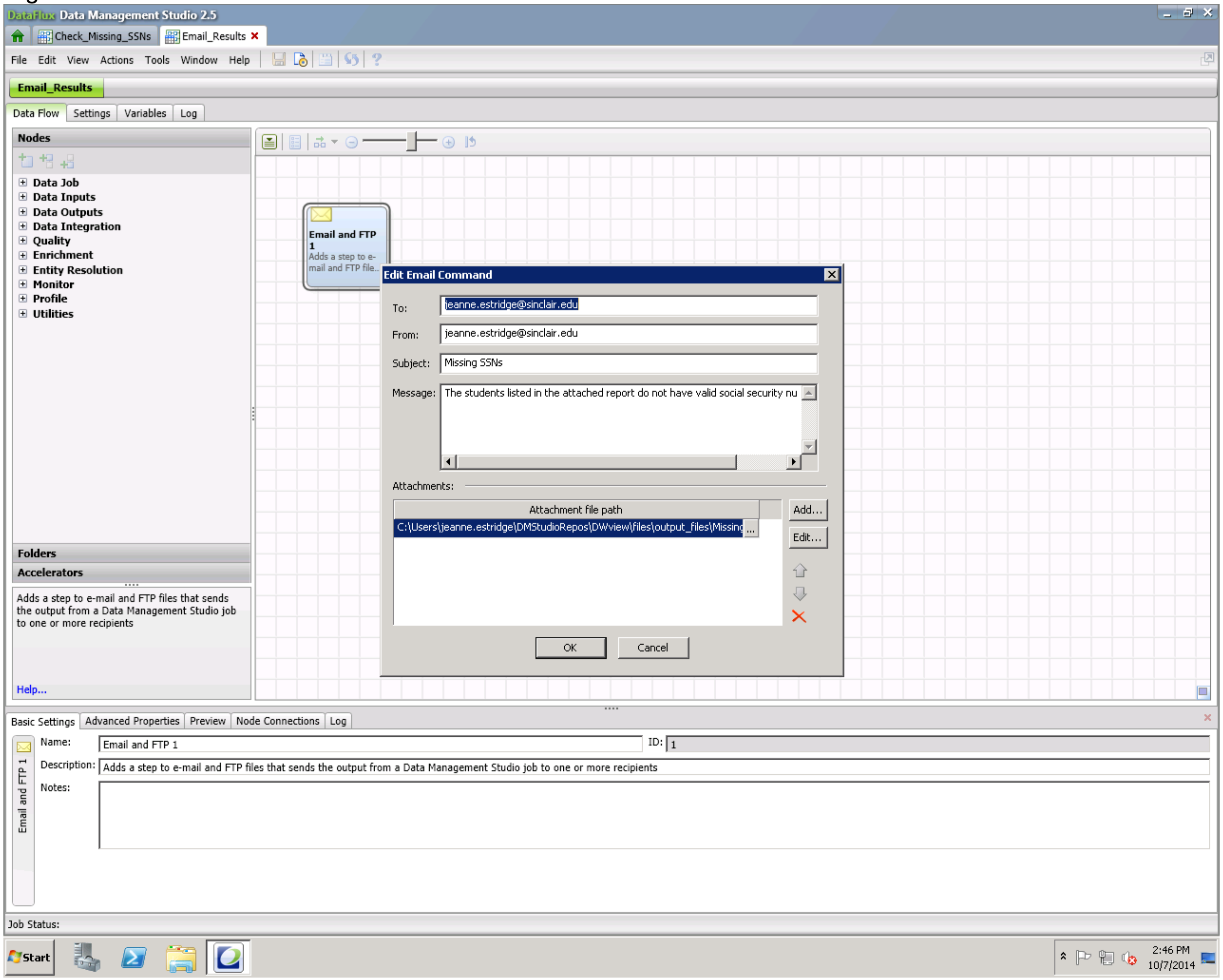
Job Status: Completed Successfully Run time: 0 hours 0 minutes 10.34 seconds

Figure 1. Create the Data Query

First, create the job that extracts the information you wish to report. The above example includes a SQL node and a text file output node, but this can be any type of data job that creates a text output file.

Once the job is built, run it to verify that it works.

Figure 2. Create the Email Job



Adds a step to e-mail and FTP files that sends the output from a Data Management Studio job to one or more recipients

To: jeanne.estridge@sinclair.edu

From: jeanne.estridge@sinclair.edu

Subject: Missing SSNs

Message: The students listed in the attached report do not have valid social security nu

Attachment file path
C:\Users\jeanne.estridge\DMStudioRepos\DWview\files\output_files\Missing

Name: Email and FTP 1

ID: 1

Description: Adds a step to e-mail and FTP files that sends the output from a Data Management Studio job to one or more recipients

Notes:

Figure 2. Create the Email Job

Next create the email job. DataFlux® requires that email nodes be standalone. Create your email job with default a distribution list, subject line, attachment file name and verbiage. Later, you can override these to make this job generic.

Once the email job is built, verify that it works.

CREATING THE PROCESS JOB

DataFlux® process jobs allow you to group data jobs and set up loops and events they also allow macro variable substitution.

Figure 3. Create the Process Job

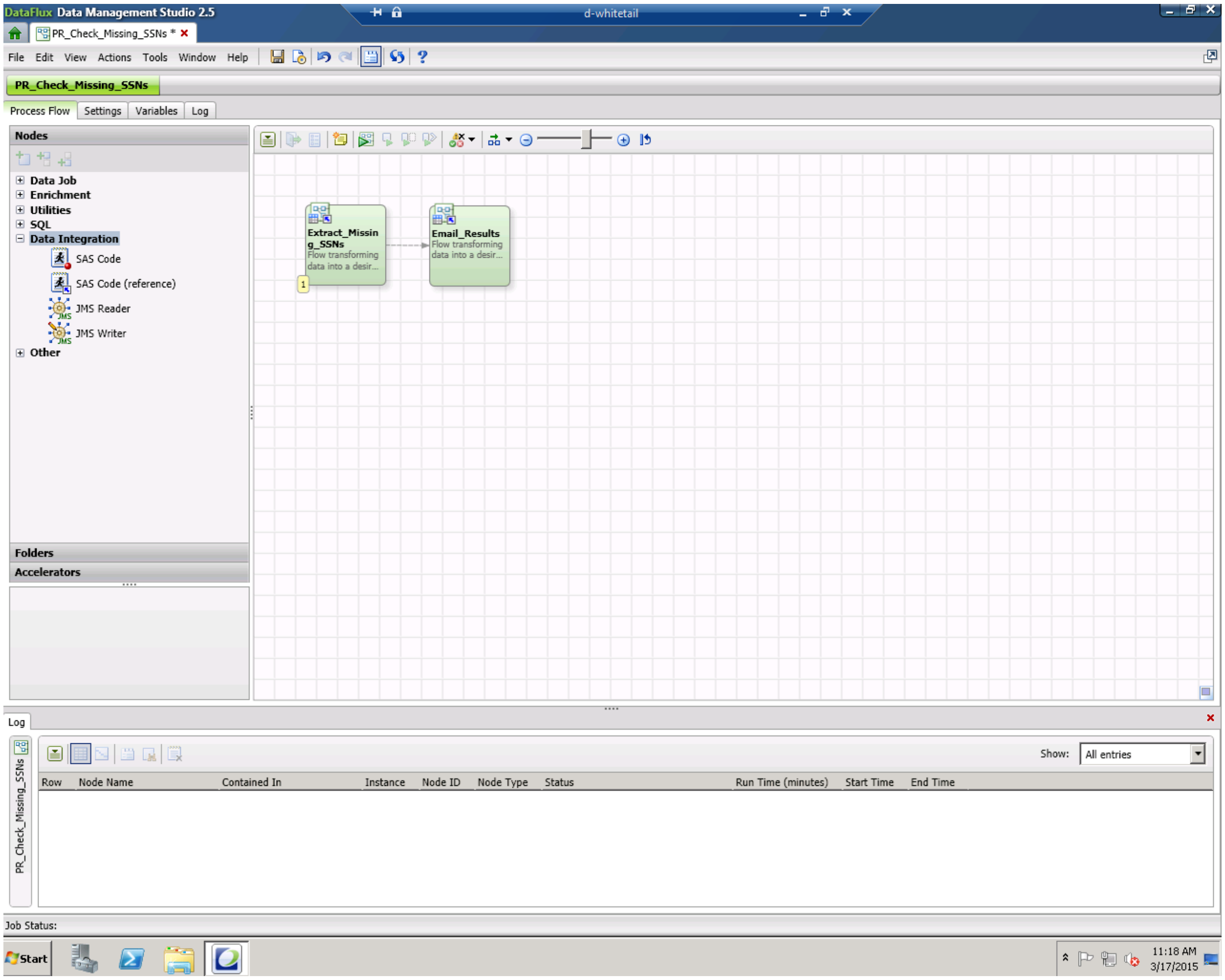


Figure 3. Create the Process Job

Next, create a process job that executes the two data jobs.

Once the job is built, verify that it works. Be sure that when you execute the process job, both data jobs run correctly.

OVERRIDING THE EMAIL JOB PARAMETERS

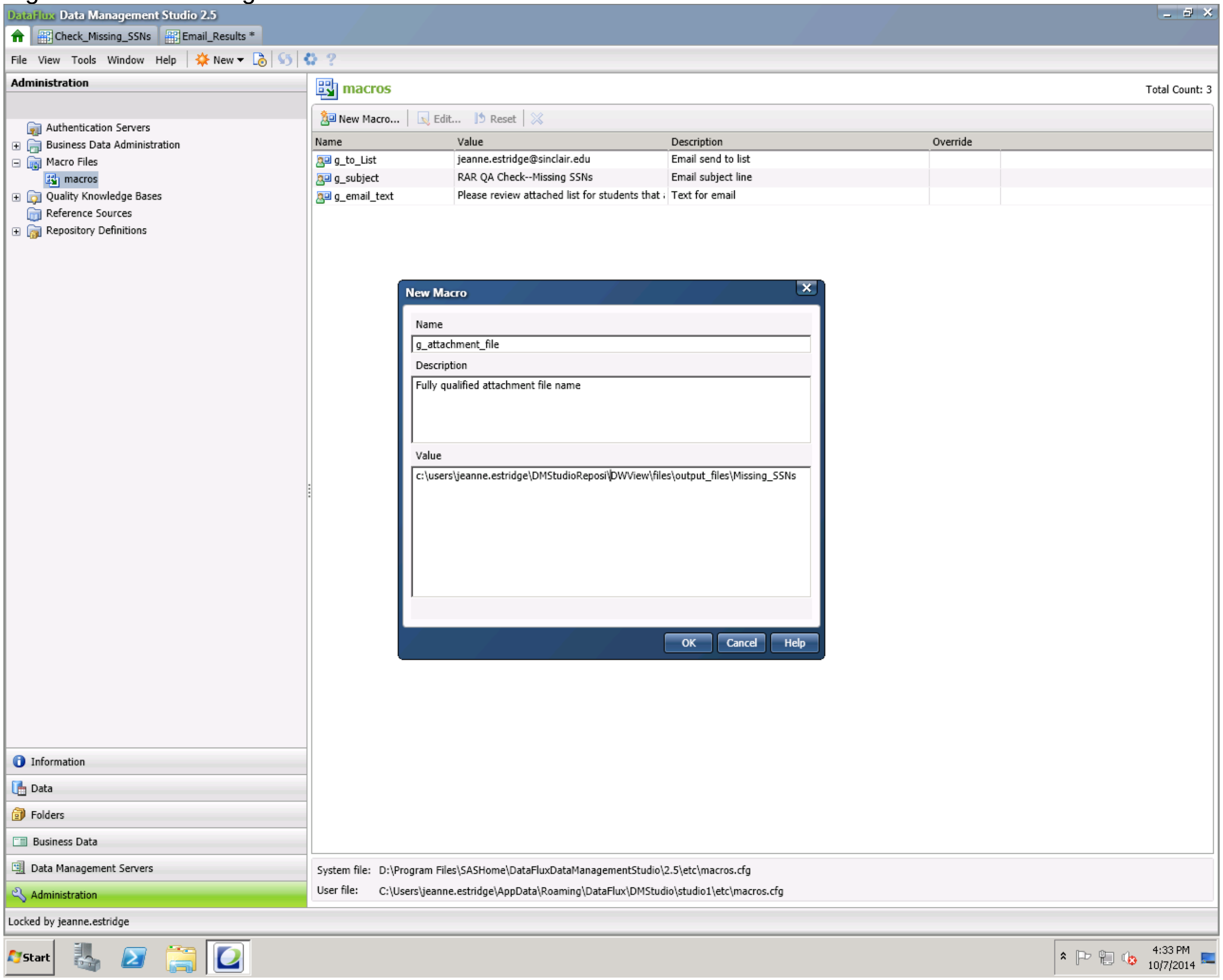
To override the email job parameters, you must first create global variables, then substitute them.

CREATE GLOBAL VARIABLES

On the Home tab, from the Administration riser bar, select Macro Files. Click on the New Macro icon and create macro variables for each of the email attributes you wish to override. Give them names that make it clear these are global variables.

1. Distribution or “to” list (g_to_list)
2. Subject (g_subject)
3. Email text (g_email_text)
4. Attachment file name (g_attachment_file)

Figure 4. Create the global variables.



Name	Value	Description	Override
g_to_List	jeanne.estridge@sinclair.edu	Email send to list	
g_subject	RAR QA Check--Missing SSNs	Email subject line	
g_email_text	Please review attached list for students that ;	Text for email	

Name
g_attachment_file

Description
Fully qualified attachment file name

Value
c:\users\jeanne.estridge\DMStudioReposi\DWView\files\output_files\Missing_SSNs

System file: D:\Program Files\SASHome\DataFluxDataManagementStudio\2.5\etc\macros.cfg
User file: C:\Users\jeanne.estridge\AppData\Roaming\DataFlux\DMStudio\studio1\etc\macros.cfg

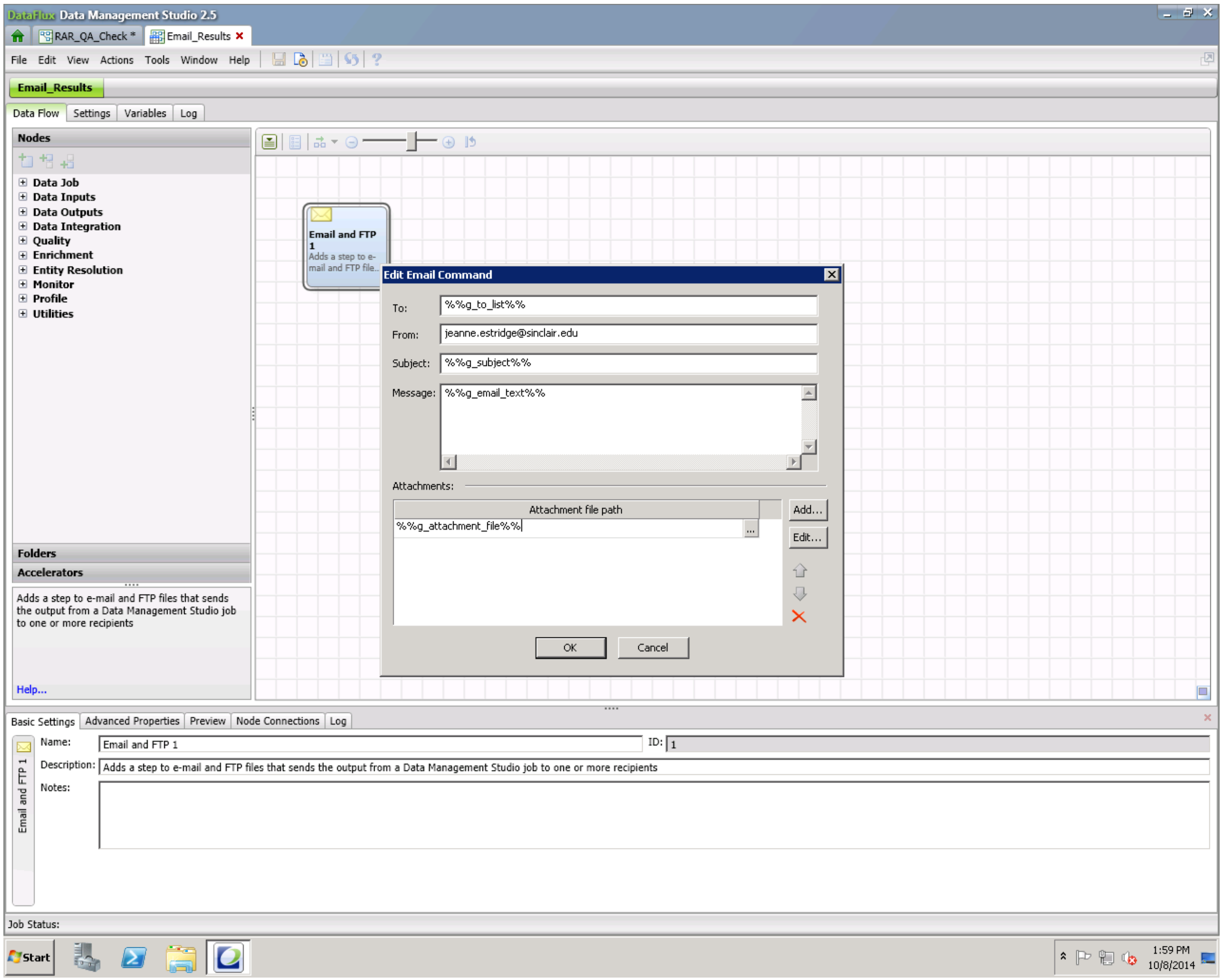
Figure 4. Create the global variables.

SUBSTITUTE GLOBAL VARIABLES FOR DEFAULTS

Update the email job to reference the global variables. Be sure to use the DataFlux® syntax of double percent signs surrounding the global variable name (e.g. %%g_to_list%%).

Verify the job still works by re-executing the process job.

Figure 5. Substitute global variables for default parameters



Adds a step to e-mail and FTP files that sends the output from a Data Management Studio job to one or more recipients

To: %%g_to_list%%

From: jeanne.estridge@sinclair.edu

Subject: %%g_subject%%

Message: %%g_email_text%%

Attachments:

Attachment file path
%%g_attachment_file%%

Name: Email and FTP 1

ID: 1

Description: Adds a step to e-mail and FTP files that sends the output from a Data Management Studio job to one or more recipients

Notes:

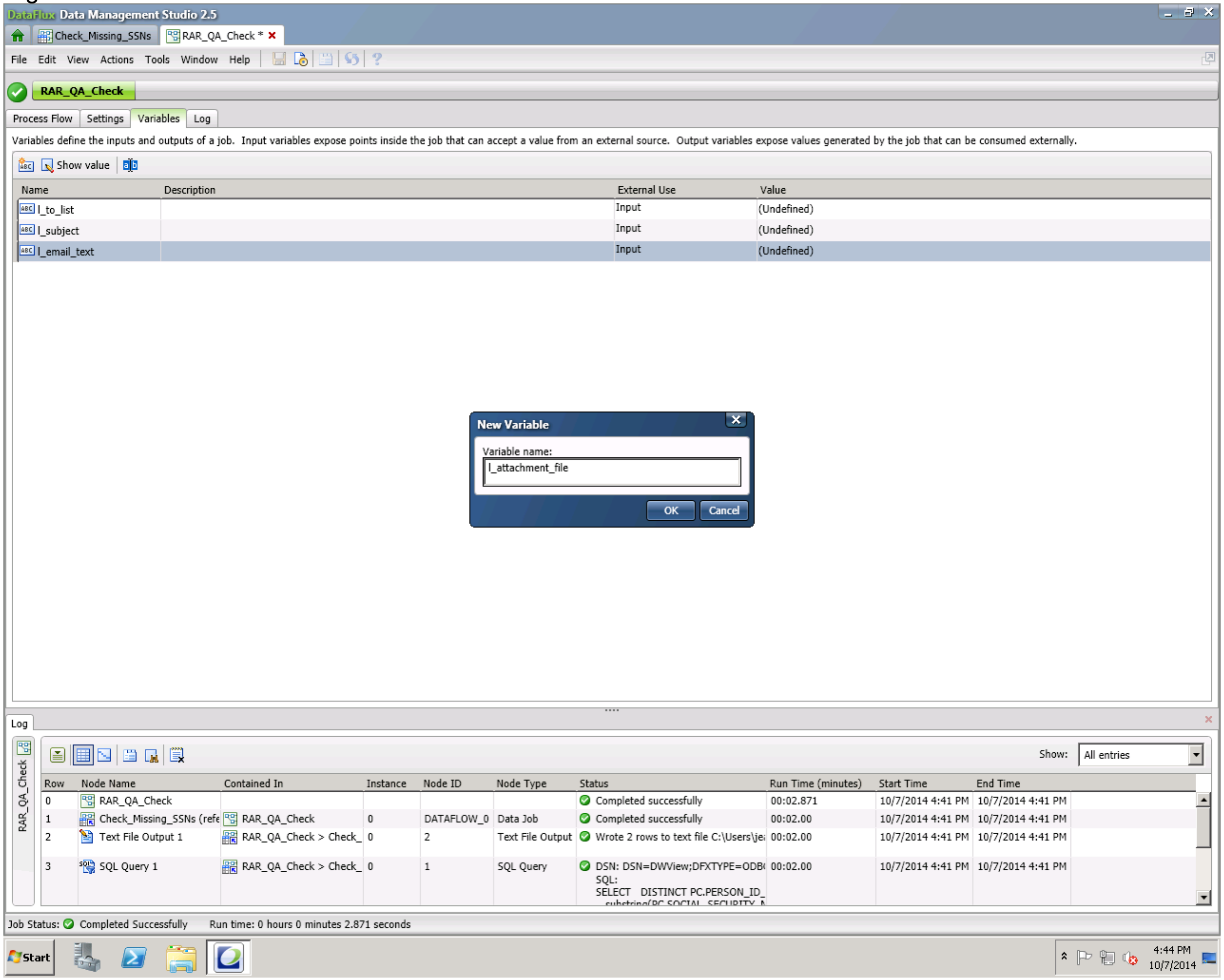
Figure 5. Substitute global variables for default parameters

CREATE LOCAL VARIABLES

Next, create local variables in your process job that correspond to the global variables.

1. Open your process job.
2. Go to the Variables tab.
3. Click on the ABC icon.
4. Enter local variables to correspond to each of the 4 macro variable names:
    a. l_to_list
    b. l_subject
    c. l_email_text
    d. l_attachment_file

Figure 6. Create local variables



DataFlux Data Management Studio 2.5
Check_Missing_SSNs
RAR_QA_Check *
File Edit View Actions Tools Window Help
RAR_QA_Check
Process Flow Settings Variables Log
Show value

Variables define the inputs and outputs of a job. Input variables expose points inside the job that can accept a value from an external source. Output variables expose values generated by the job that can be consumed externally.

Name	Description	External Use	Value
l_to_list		Input	(Undefined)
l_subject		Input	(Undefined)
l_email_text		Input	(Undefined)

Variable name: l_attachment_file

Show: All entries

Row	Node Name	Contained In	Instance	Node ID	Node Type	Status	Run Time (minutes)	Start Time	End Time
0	RAR_QA_Check					Completed successfully	00:02.871	10/7/2014 4:41 PM	10/7/2014 4:41 PM
1	Check_Missing_SSNs (refe	RAR_QA_Check	0	DATAFLOW_0	Data Job	Completed successfully	00:02.00	10/7/2014 4:41 PM	10/7/2014 4:41 PM
2	Text File Output 1	RAR_QA_Check > Check_	0	2	Text File Output	Wrote 2 rows to text file C:\Users\je	00:02.00	10/7/2014 4:41 PM	10/7/2014 4:41 PM
3	SQL Query 1	RAR_QA_Check > Check_	0	1	SQL Query	DSN: DSN=DWView;DFXTYPE=ODB SQL: SELECT DISTINCT PC.PERSON_ID_ substring(PC.SOCIAL_SECURITY_N	00:02.00	10/7/2014 4:41 PM	10/7/2014 4:41 PM

Job Status: Completed Successfully   Run time: 0 hours 0 minutes 2.871 seconds

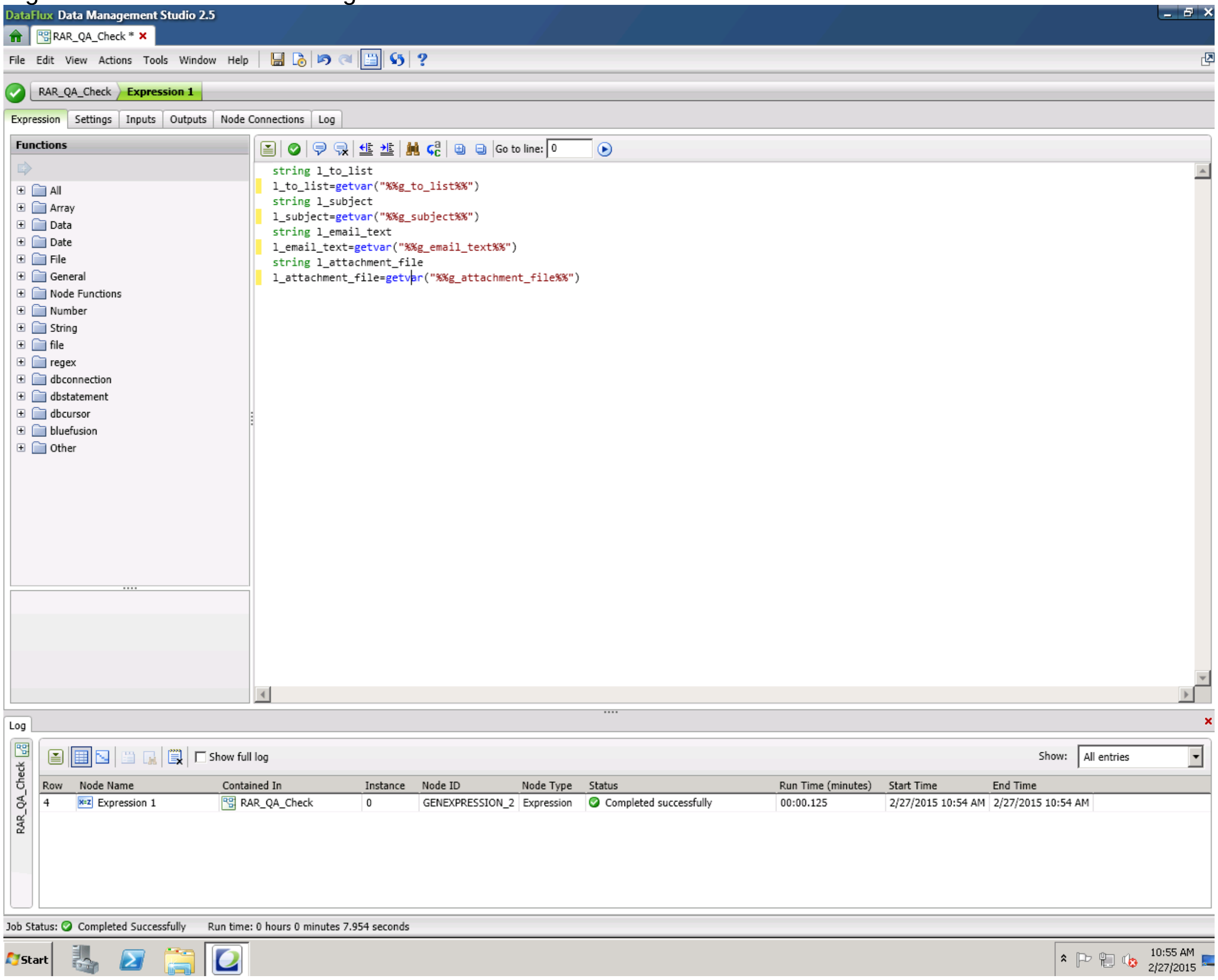
Figure 6. Create local variables

ASSOCIATE LOCAL AND GLOBAL VARIABLES

Next, associate the local variables with the global variables.

1. Open your process job.
2. Click on the process flow tab.
3. Expand the Utilities node list
4. Double-click on “Expression”
5. In the expression pane, enter the associations as follows:

Figure 7. Associate local and global variables



```
string l_to_list
l_to_list=getvar("%%g_to_list%%")
string l_subject
l_subject=getvar("%%g_subject%%")
string l_email_text
l_email_text=getvar("%%g_email_text%%")
string l_attachment_file
l_attachment_file=getvar("%%g_attachment_file%%")
```

☐ Show full log   Show: All entries

Row	Node Name	Contained In	Instance	Node ID	Node Type	Status	Run Time (minutes)	Start Time	End Time
4	Expression 1	RAR_QA_Check	0	GENEXPRESSION_2	Expression	Completed successfully	00:00.125	2/27/2015 10:54 AM	2/27/2015 10:54 AM

Job Status: Completed Successfully   Run time: 0 hours 0 minutes 7.954 seconds

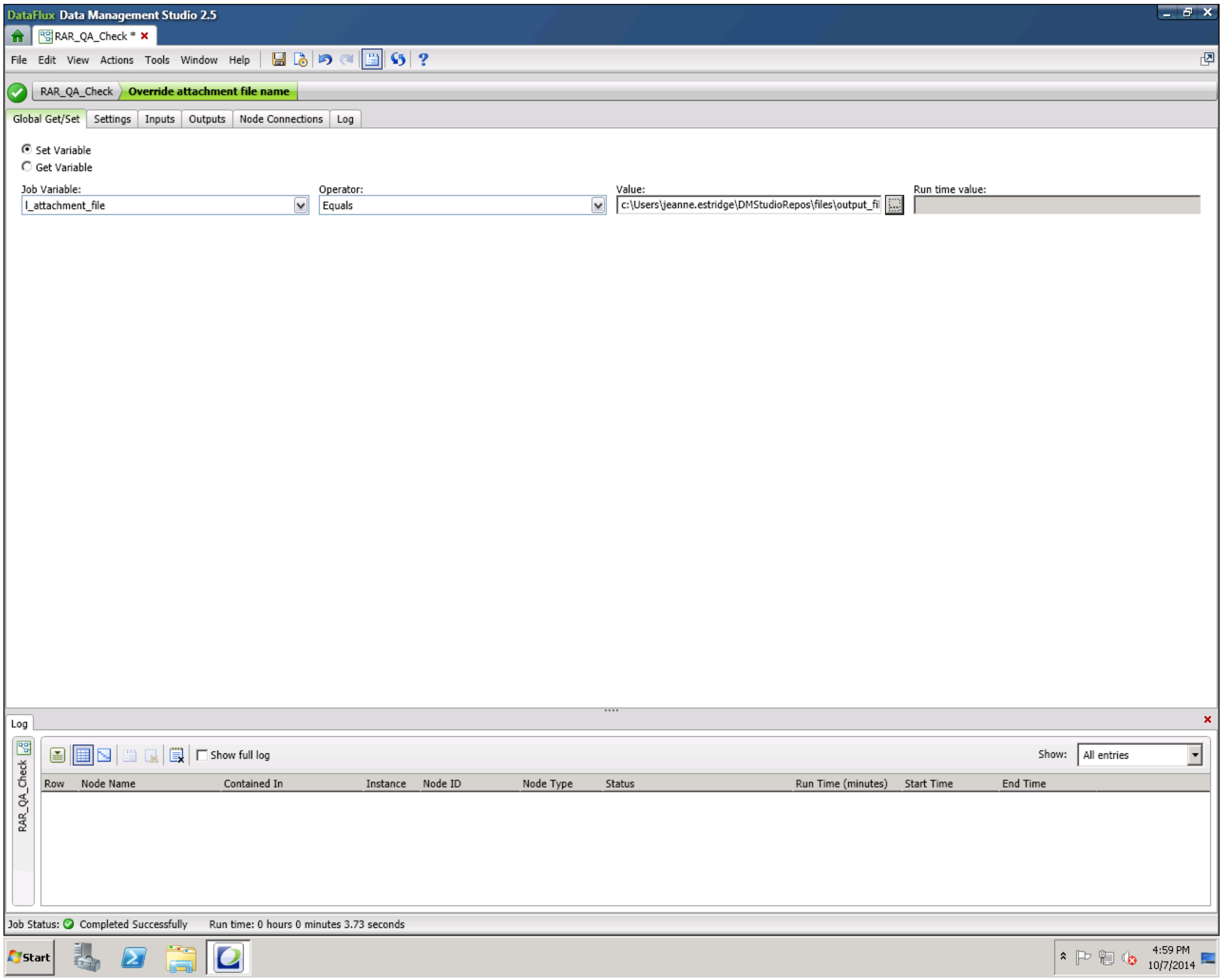
Figure 7. Associate local and global variables

OVERRIDE GLOBAL VARIABLES

Next, you’ll want to override the global variables.

1. Return to the process job.
2. Click on Utilities.
3. Add a Global Get/Set Node.
4. Double-click to open the Get/Set node.
5. Select a local variable from the dropdown list.
6. Select “equals.”
7. Enter the override value.
8. Do this for each variable you wish to override.
9. Be sure to rearrange the job so that the email node executes after the overrides.

Figure 8. Override global variables



● Set Variable
○ Get Variable

Job Variable: l_attachment_file

Operator: Equals

Value: c:\Users\jeanne.estridge\DMStudioRepos\files\output_fil

Run time value:

RAR_QA_Check

□ Show full log

Show: All entries

Row	Node Name	Contained In	Instance	Node ID	Node Type	Status	Run Time (minutes)	Start Time	End Time

Job Status: Completed Successfully   Run time: 0 hours 0 minutes 3.73 seconds

4:59 PM
10/7/2014

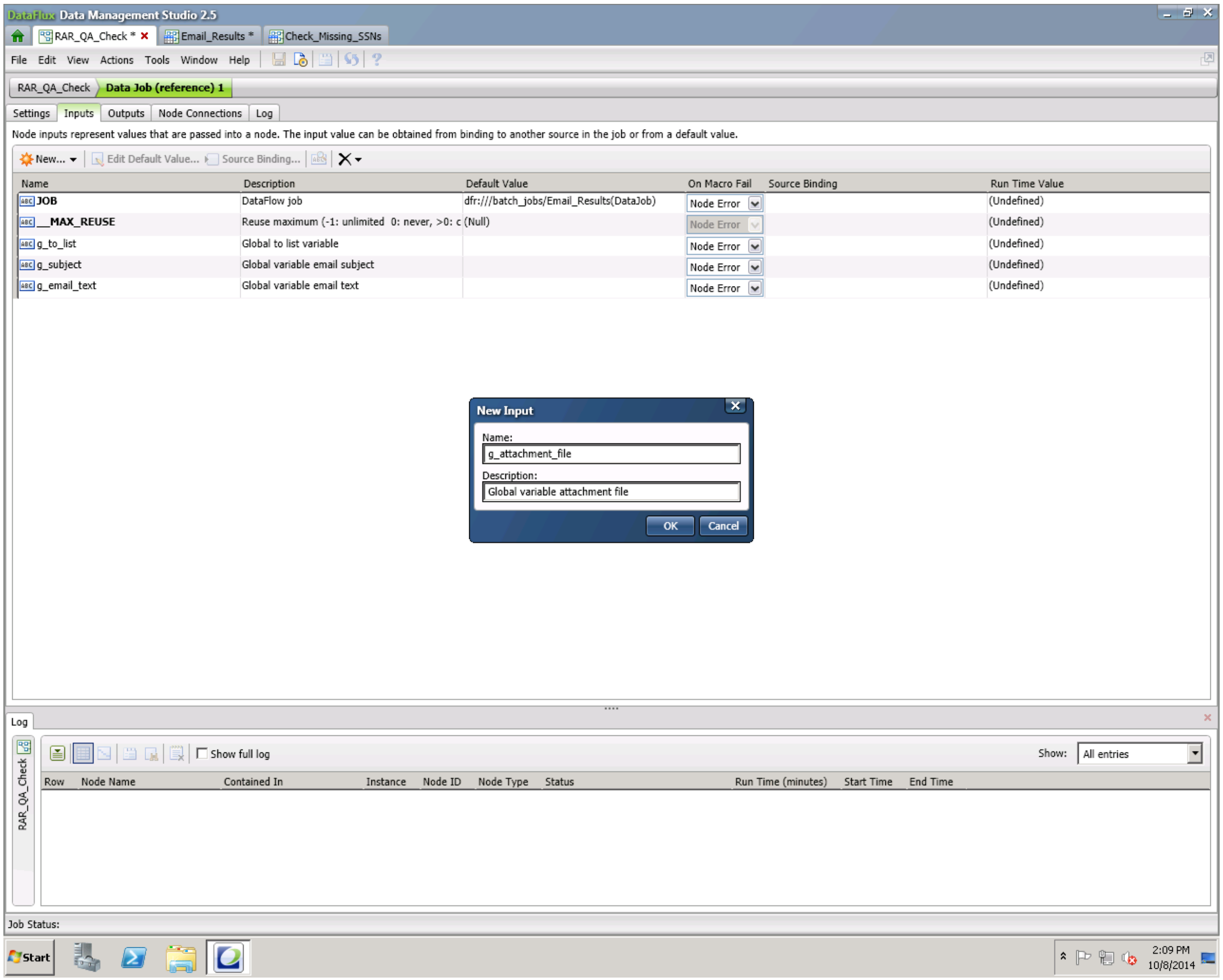
Figure 8. Override global variables

SOURCE BIND THE VARIABLES

Next, you'll create input strings for source binding:

1. Return to the process job.
2. Rick click on the email job.
3. Select "Properties."
4. Click on the inputs tab.
5. Click on New
6. Select New Input String
7. Enter the global variable and then click ok.

Figure 9. Create input strings for source binding


DataFlux Data Management Studio 2.5
RAR_QA_Check *
Email_Results *
Check_Missing_SSNs
File Edit View Actions Tools Window Help
RAR_QA_Check
Data Job (reference) 1
Settings Inputs Outputs Node Connections Log
New... Edit Default Value... Source Binding...

Node inputs represent values that are passed into a node. The input value can be obtained from binding to another source in the job or from a default value.

Name	Description	Default Value	On Macro Fail	Source Binding	Run Time Value
JOB	DataFlow job	dfr:///batch_jobs/Email_Results(DataJob)	Node Error		(Undefined)
__MAX_REUSE	Reuse maximum (-1: unlimited 0: never, >0: c	(Null)	Node Error		(Undefined)
g_to_list	Global to list variable		Node Error		(Undefined)
g_subject	Global variable email subject		Node Error		(Undefined)
g_email_text	Global variable email text		Node Error		(Undefined)

New Input

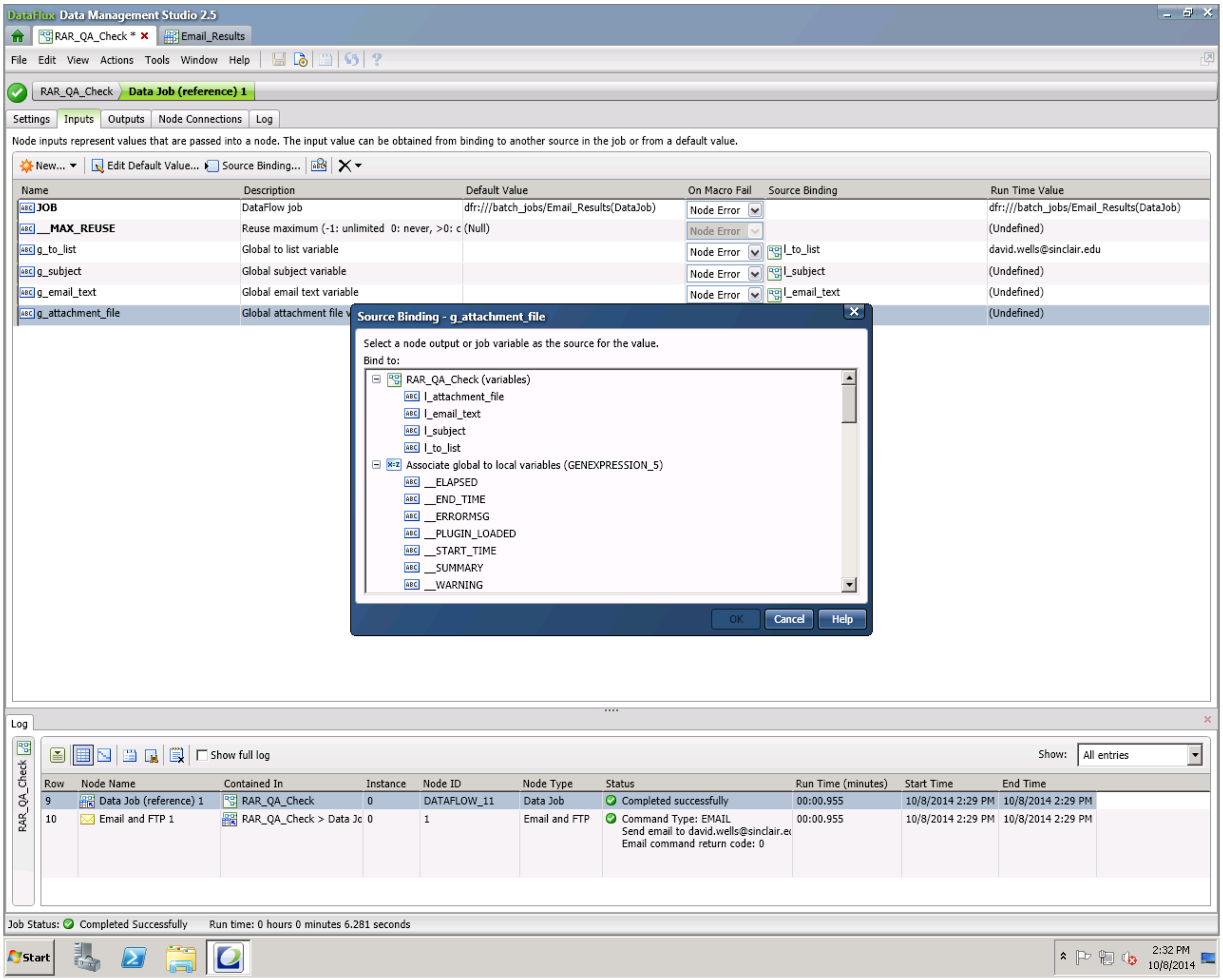
Name: g_attachment_file

Description: Global variable attachment file

OK
Cancel
Log
RAR_QA_Check
□ Show full log
Show: All entries
Row Node Name Contained In Instance Node ID Node Type Status Run Time (minutes) Start Time End Time
Job Status:
Start
2:09 PM
10/8/2014

Figure 9. Create input strings for source binding

Figure 10. Source Bind Variables



DataFlux Data Management Studio 2.5
RAR_QA_Check *
Email_Results
File Edit View Actions Tools Window Help
RAR_QA_Check
Data Job (reference) 1
Settings Inputs Outputs Node Connections Log

Node inputs represent values that are passed into a node. The input value can be obtained from binding to another source in the job or from a default value.

New... Edit Default Value... Source Binding...

Name	Description	Default Value	On Macro Fail	Source Binding	Run Time Value
JOB	DataFlow job	dfr:///batch_jobs/Email_Results(DataJob)	Node Error		dfr:///batch_jobs/Email_Results(DataJob)
__MAX_REUSE	Reuse maximum (-1: unlimited 0: never, >0: c	(Null)	Node Error		(Undefined)
g_to_list	Global to list variable		Node Error	l_to_list	david.wells@sinclair.edu
g_subject	Global subject variable		Node Error	l_subject	(Undefined)
g_email_text	Global email text variable		Node Error	l_email_text	(Undefined)
g_attachment_file	Global attachment file v				(Undefined)

Source Binding - g_attachment_file

Select a node output or job variable as the source for the value.

Bind to:

- RAR_QA_Check (variables)
  - l_attachment_file
  - l_email_text
  - l_subject
  - l_to_list
- Associate global to local variables (GENEXPRESSION_5)
  - __ELAPSED
  - __END_TIME
  - __ERRORMSG
  - __PLUGIN_LOADED
  - __START_TIME
  - __SUMMARY
  - __WARNING

Row	Node Name	Contained In	Instance	Node ID	Node Type	Status	Run Time (minutes)	Start Time	End Time
9	Data Job (reference) 1	RAR_QA_Check	0	DATAFLOW_11	Data Job	Completed successfully	00:00.955	10/8/2014 2:29 PM	10/8/2014 2:29 PM
10	Email and FTP 1	RAR_QA_Check > Data Jc	0	1	Email and FTP	Command Type: EMAIL Send email to david.wells@sinclair.ec Email command return code: 0	00:00.955	10/8/2014 2:29 PM	10/8/2014 2:29 PM

Job Status: Completed Successfully   Run time: 0 hours 0 minutes 6.281 seconds

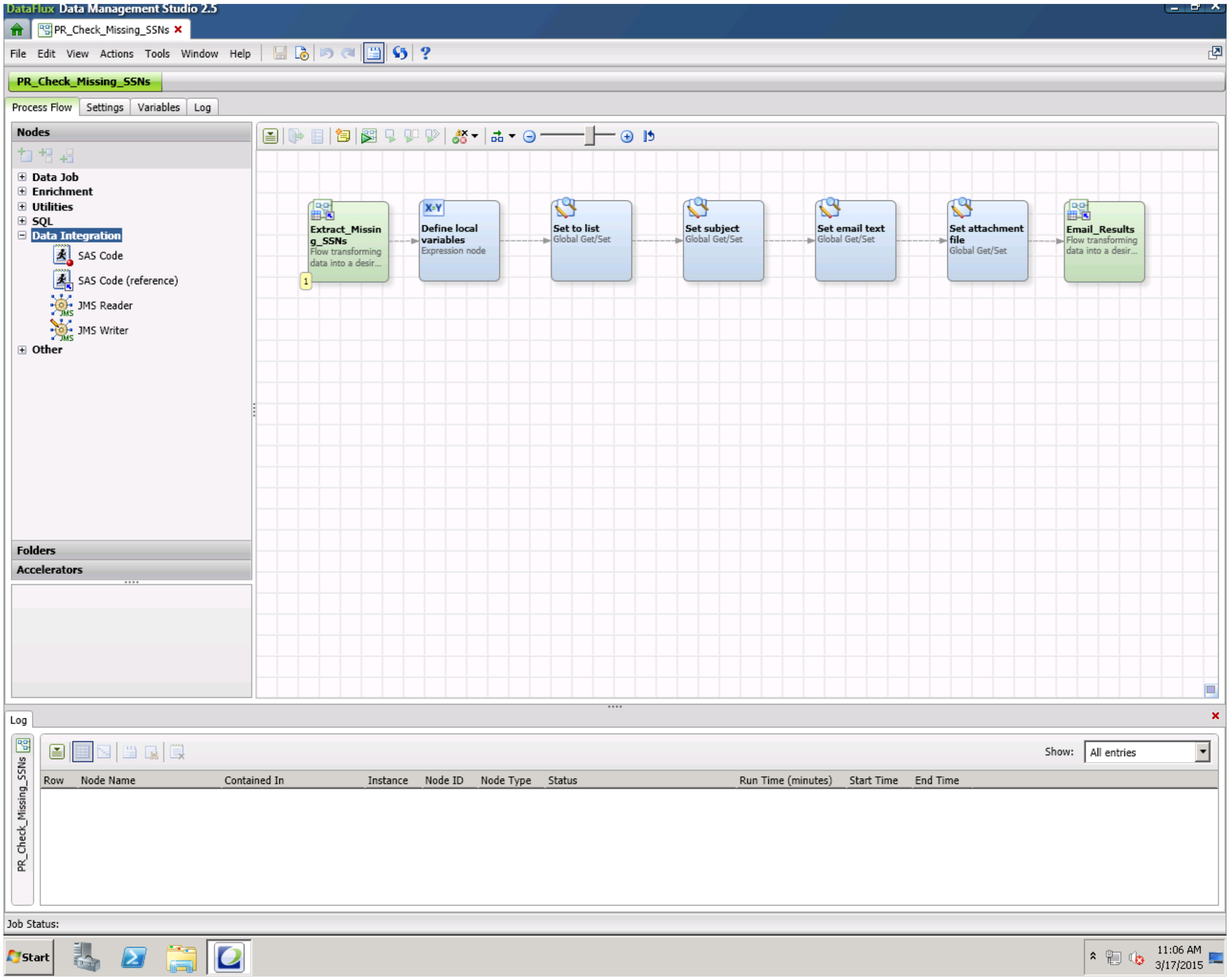
Figure 10. Source Bind Variables

Next, you'll source bind the variables:

1) Click on a global variable name.
2) Click on source binding.
3) Click on the appropriate matching local variable.
4) Click ok.
5) Run the entire process job to verify that it still works and that the local variables override the global variables.

Now, if you run the email node standalone, it will use the global variables. If you run it from the process job, it will use the local variables.

Figure 11. Final Job



Row	Node Name	Contained In	Instance	Node ID	Node Type	Status	Run Time (minutes)	Start Time	End Time

Figure 11. Final Job

When you’re finished, your final job will look something like the above figure.

CALLING THE JOB FROM THE COMMAND PROMPT

Create a .bat file that includes the following statement:

```
call dmpexec
-j c:\Users\jeanne.estridge\DMStudioRepos\DWView\files\process_services\RAR_QA_Check.djf
-l c:\temp\RARLog.txt
-o "BASE/REPOS_FILE_ROOT=c:\Users\jeanne.estridge\DMStudioRepos\DWView\files"
-o "g_to_list=jeanne.estridge@sinclair.edu"
-o "g_subject=Data Quality Check—Missing SSNs"
-o “g_attachment_file=C:\Users\your.name\DMStudioRepos\DWview\files\output_files\Missing_SSNs.txt"
-o "g_email_text=The attached file contains a list of students with missing social security numbers,"
```

Since this job cannot access the global macro variables inside DMStudio, be sure to include an override for every variable you have defined.

Note: If you set this up to run from DMServer, rather than DMStudio, you won’t need to do this unless you choose to.

This job just execute from the directory where dmpexec resides.

Verify that the call executes the job properly.

Once you’ve tested this out, you can use your job scheduling software to schedule the bat file to run.

For more information on executing Dataflux jobs from the command line, see the Command Line Usage Notes at: http://support.sas.com/documentation/onlinedoc/dfdmstudio/2.6/dmpdmsug/Content/dfU_T_DataJob_RunBatch.html

CONCLUSION

While DataFlux® is primarily designed to be used in an interactive mode, it is possible to write parameterized data quality jobs that run on a scheduled basis and notify the appropriate personnel when quality problems are diagnosed.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Jeanne Estridge
Sinclair Community College
Jeanne.estridge@sinclair.edu