SAS® GLOBAL FORUM 2016

IMAGINE. CREATE. INNOVATE.

A Report Tracking Framework

- Using SAS® to Report Status of Scheduled Jobs

ePoster 11540

Jia Heng
Wyndham Destination Network

#SASGF

A Report Tracking Framework: Using SAS® to Report Status of Schedule Jobs

Jia Heng

Wyndham Destination Network

ABSTRACT

With increasing data needs, it becomes more and more unwieldy to ensure that all scheduled jobs are running successfully and on time. Worse, maintaining your reputation as an information provider becomes a precarious prospect as the likelihood increases that your customers alert you of reporting issues before you are even aware yourself. By combining various SAS® capabilities and tying them together with concise visualizations, it is possible to track jobs actively and alert customers of issues before they become a problem. This paper introduces a report tracking framework that helps achieve this goal and improve customer satisfaction. The report tracking starts by obtaining table and job statuses and then color-codes them by severity levels. Based on the job status, it then goes deeper into the log to search for potential errors or other relevant information to help assess the processes. The last step is to send proactive alerts to users informing them of possible delays or potential data issues.

DATA PREPARATION

The report tracking mechanism introduced in this paper begins with collecting information related to scheduled jobs. A good understanding of what jobs you manage is essential to accurately reporting the job status. A dataset containing the job list needs to be prepared beforehand (This list also requires ongoing maintenance effort for updating when new jobs come in or old jobs retire). The following fields are recommended:

- Log names: They can be found in SAS® server "Files" folder. Depending on how the jobs are set up, log_name is usually name of the whole job flow followed by name of the specific job within that flow
- Frequency, Day, and Time are to follow specific format to calculate the due datetime of the jobs/datasets later

  Frequency: Can be Day/week/month/year

  DayOf: Day of the week/month/year when job is due

  TimeOf: Time of the day when job is due. Format hhmm8. is suggested
- Category: Datasets and other jobs are treated differently. For jobs that generate dashboard and reports, log run time is used to determine status. For datasets, we can go one step further to get modify time
- Output: Key information for users

Table 1. Example of Job List

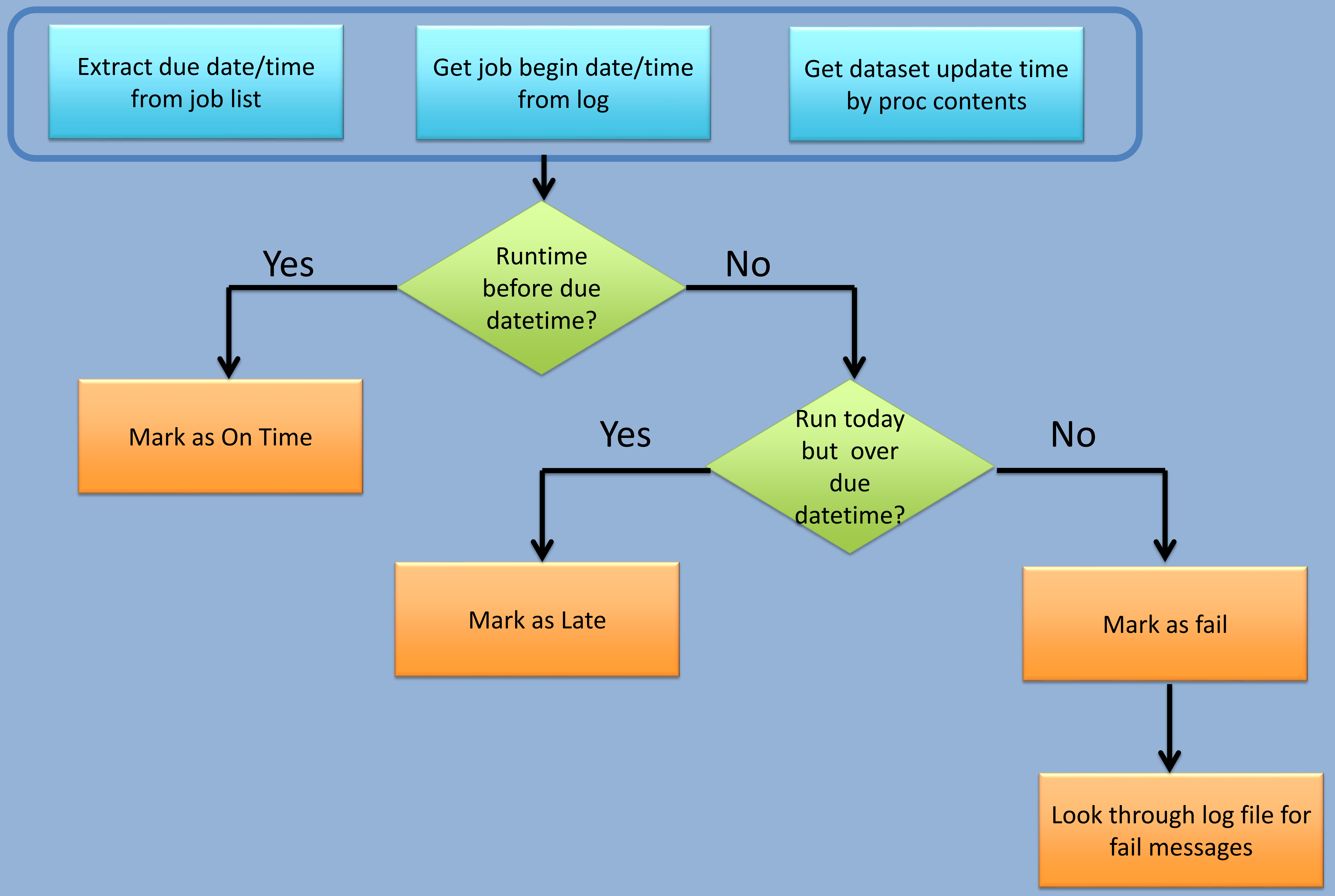
Category	Job_name	Log_name	Output	Frequency	DayOf	TimeOf
Dashboard	sas_dashboard_job1	sas_dashboard_flow1_sas_dashboard_job1	Dashboard output 1	Day	0	6:00:00
Dashboard	sas_dashboard_job2	sas_dashboard_flow1_sas_dashboard_job2	Dashboard output 2	Day	0	6:00:00
Dashboard	sas_dashboard_job3	sas_dashboard_flow2_sas_dashboard_job3	Dashboard output 3	Week	2	18:00:00
Reports	sas_report_job1	sas_report_flow1_sas_report_job1	Email PDF attach	Month	8	14:00:00
Reports	sas_report_job2	sas_report_flow2_sas_report_job2	Email Excel attach	Week	2	8:00:00
Datasets	sas_dataset_job1	sas_dataset_flow1_sas_dataset_job1	Dataset1	Week	0	8:00:00
Datasets	sas_dataset_job2	sas_dataset_flow1_sas_dataset_job2	Dataset2	Week	0	8:00:00

METHOD (PROCESS FLOW)

The process starts with getting all the measures for decision making. Two key measures are: due time of the job and job run time. Update time of datasets is also obtained if a dataset is the output.

Once all information is present, a decision is made following the decision tree below. The criteria for deciding if the jobs are on time/late/fail is dependent on different company rules. The chart below should be considered as a general guidance. Some assumptions need to be made. In this paper, if the job is not delivered when tracking process is completed, it is considered as fail. It doesn't necessarily mean the job won't run later.

At the same time, error messages are retrieved from log file and appended to the jobs/datasets that are considered as fail. Finally, an alert is sent out to the reporting management team and users.



Extract due date/time from job list

Get job begin date/time from log

Get dataset update time by proc contents

Runtime before due datetime?

Yes

No

Mark as On Time

Run today but over due datetime?

Yes

No

Mark as Late

Mark as fail

Look through log file for fail messages

A Report Tracking Framework: Using SAS® to Report Status of Schedule Jobs

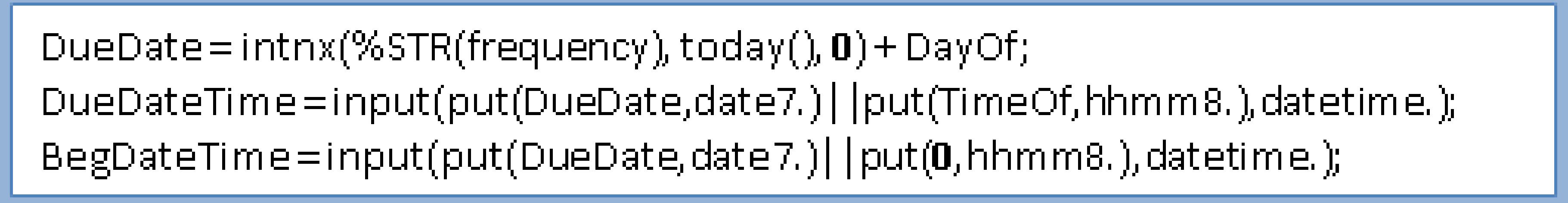
Jia Heng

Wyndham Destination Network

METHOD (DETAILED CODING)

Extract due date/time from job list

Three fields Frequency, DayOf and TimeOf in the pre-compiled job list are used for getting due date/time. Another field beg_datetime is used to mark beginning of the day to make sure we do not mistakenly compare to the time of previous run.



```
DueDate=intnx(%STR(frequency), today(), 0)+DayOf;
DueDateTime=input(put(DueDate,date7.)||put(TimeOf,hhmm8.),datetime.);
BegDateTime=input(put(DueDate,date7.)||put(0,hhmm8.),datetime.);
```

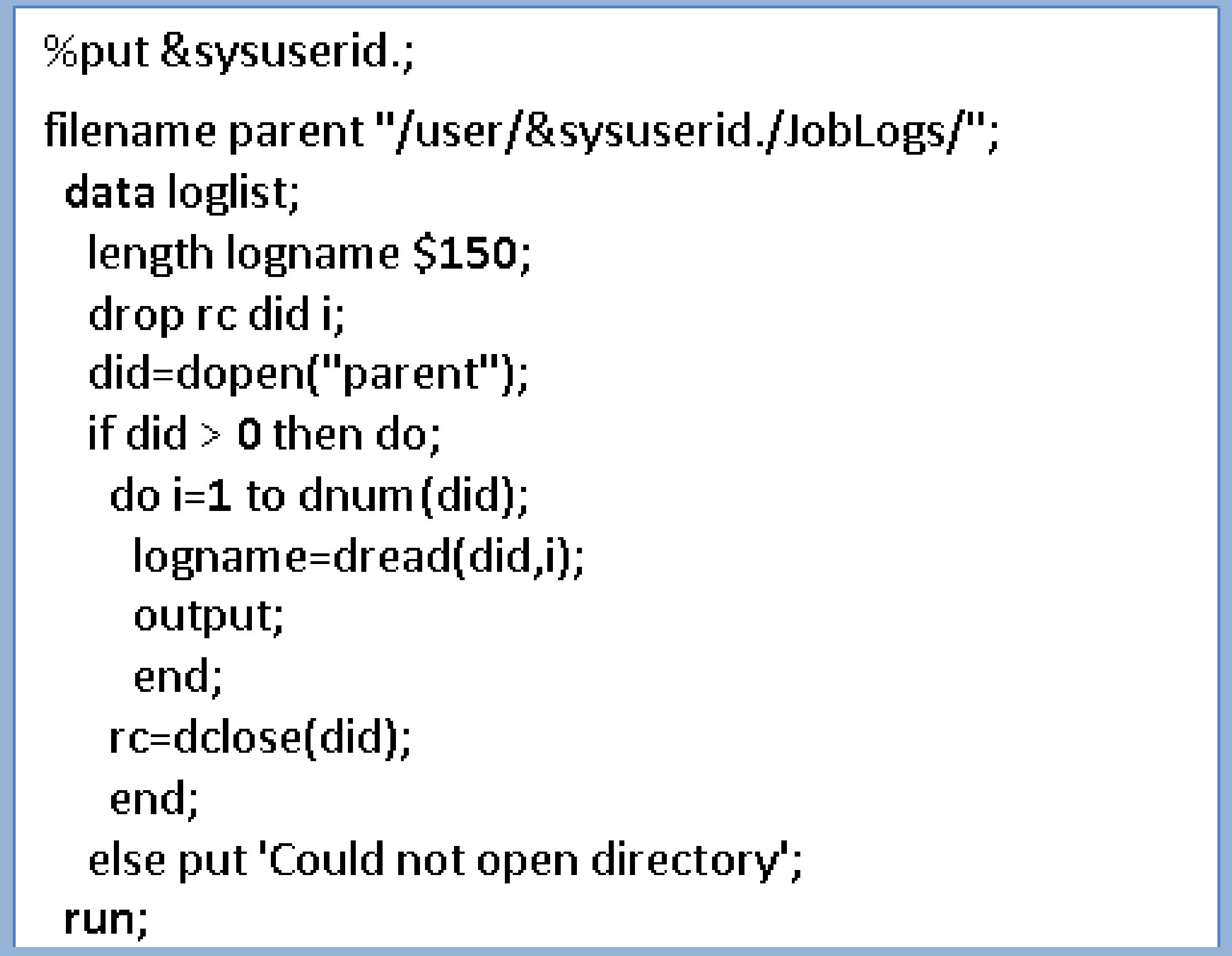
Get job begin date/time from log

This step obtains all logs that currently exist on the server. Log name is usually in the format of

logname_yyyy.mm.dd_hh.mm.ss.log

Substr() function can be used to extract the runtime of the log.

In most cases, the server keeps multiple log files for one scheduled job. After extracting the dates from logs, retain only the most recent log of jobs.



```
%put &sysuserid.;
filename parent "/user/&sysuserid./JobLogs/";
 data loglist;
  length logname $150;
  drop rc did i;
  did=dopen("parent");
  if did > 0 then do;
   do i=1 to dnum(did);
    logname=dread(did,i);
    output;
    end;
   rc=dclose(did);
   end;
  else put 'Could not open directory';
 run;
```

Get dataset update time by proc contents

A loop is set up to run proc contents on each of the dataset names in the job list and append the output field “modate” as the dataset’s last updated date onto the job list.

Runtime before due datetime?

By comparing the run_time and due_time, status of the job is assigned:

On Time: BegDateTime < runtime <= DueDatetime

Late: DueDateTime < runtime < Now

Fail: runtime < BegDateTime or runtime is null

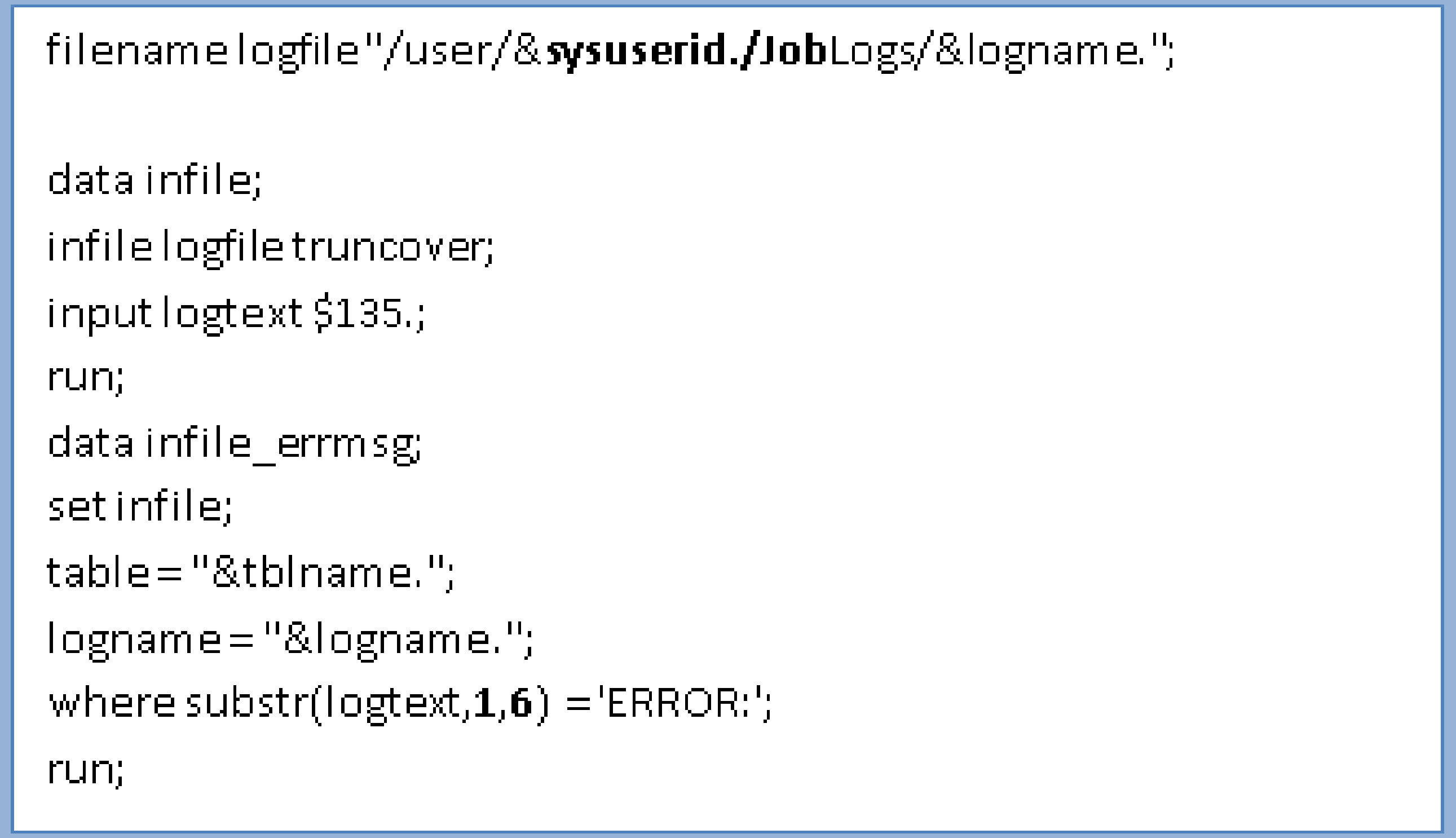
METHOD (DETAILED CODING)

Look through log file for fail messages

Each log file can be retrieved as a dataset with one text line per row.

With code on the left, only the rows with text “error” are picked up. For simplified results, some system generated errors which are not pertinent to the issue, can be eliminated, eg. “ERROR: Errors printed on page18”.

Again, A loop can be used to append all error related messages to the log files.



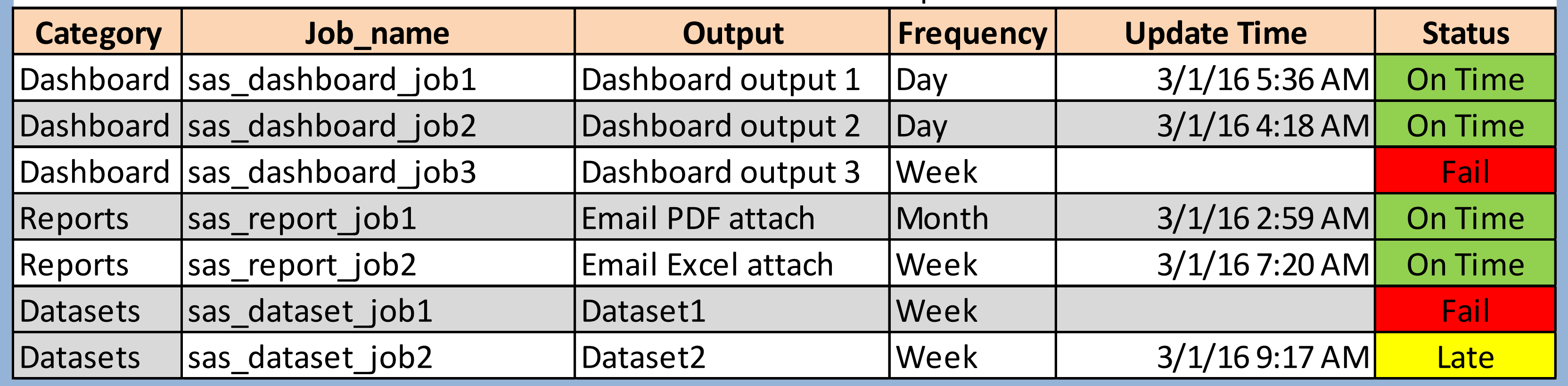
```
filename logfile "/user/&sysuserid./JobLogs/&logname.";

data infile;
infile logfile truncover;
input logtext $135.;
run;
data infile_errmsg;
set infile;
table="&tblname.";
logname="&logname.";
where substr(logtext,1,6) ='ERROR:';
run;
```

DISPLAY OF RESULTS

There are different ways of displaying the results. Summary table showing number of jobs failing in each of the category is one way if there are more than a hundred jobs to manage. Displaying the whole job list with the use of conditional formatting to allow users to pin point problem areas at first glance of the report is another way. Table two is an example of simple output with conditional formatting based on value of status.

Table 2. Job Status Output



Category	Job_name	Output	Frequency	Update Time	Status
Dashboard	sas_dashboard_job1	Dashboard output 1	Day	3/1/16 5:36 AM	On Time
Dashboard	sas_dashboard_job2	Dashboard output 2	Day	3/1/16 4:18 AM	On Time
Dashboard	sas_dashboard_job3	Dashboard output 3	Week		Fail
Reports	sas_report_job1	Email PDF attach	Month	3/1/16 2:59 AM	On Time
Reports	sas_report_job2	Email Excel attach	Week	3/1/16 7:20 AM	On Time
Datasets	sas_dataset_job1	Dataset1	Week		Fail
Datasets	sas_dataset_job2	Dataset2	Week	3/1/16 9:17 AM	Late

CONCLUSION

With the setup of an automated alert process, the tracking procedure can be greatly simplified and efficient. In addition, by knowing the data issues in advance, it helps the team improve data integrity and customer satisfaction.

SAS® GLOBAL FORUM 2016

IMAGINE. CREATE. INNOVATE.

LAS VEGAS | APRIL 18-21

#SASGF