

Automatic Access Control via SAS: More Efficient and Smart

Yuichi Koretaka, SHIONOGI & CO., LTD., Japan
Masakazu Fujiwara, SHIONOGI & CO., LTD., Japan
Yuki Yoshida, SHIONOGI & CO., LTD., Japan
Yoshitake Kitanishi, SHIONOGI & CO., LTD., Japan

ABSTRACT

A wide variety of information is accumulated during clinical trials. In Shionogi & Co. Ltd., as is usual with most of the pharmaceutical companies, the accumulated information is stored as electronic data at specified folders of a server attached to an in-house database system (managed by the Biostatistics Department), considering the property of the information. Development of a drug during clinical phase is carried by a project team consisted of persons from various clinical related departments. Hence, members of Biostatistics Department are allocated to the development project team as ones of data analysis team, and can access all the folders in the server. Though, some of the information needs confidential treatment. As for high confidential information (e.g. clinical study report, clinical data, programs and analysis results etc.), the access rights should be given to limited members who have roles deemed appropriate to access respective information (or folders). On the other hand, other information thought to be not so high in confidentiality (e.g. analysis plan, analysis results (i.e. TLFs), study protocol and etc.), should be shared so that the knowledge can be effectively used among the members of the department. Thus, we decided to control access rights of the members of each analysis team to each folder of the server, considering the roles of the team and the property of information. However, folder structure of our server is very complicated, because it is designed in reference to Study Data Specifications that FDA released in 2012 [1]. Therefore, we are afraid that setting access rights manually is inefficient and might cause human error. To solve these problems, we developed SAS® programs that automatically allocate access rights to each of the persons involved. The purpose of this paper is to describe our information management process including SAS programs and provide some important snippets of SAS code. Note that we use SAS 9.2 and Microsoft Office 2010 on Windows server 2008 R2 standard.

INTRODUCTION

Information management is a challenge because many factors (e.g. insider trading, transparency, knowledge sharing, security and etc.) are related to it. In particular, compatibility of prevention of insider trading and assurance of knowledge sharing seems difficult. Thus, to minimize the company risks, rigorous management of information is necessary. Microsoft website [2] said that 'A user group is a collection of user accounts that all have the same security rights. User groups are also sometimes referred to as security groups'. Therefore, information management by controlling access rights of a user group seems reasonable. Hence, we developed SAS programs for controlling access rights. The reason to select SAS is that SAS has linkage functions with Excel and Command Prompt, and can handle datasets more flexibly than other software. By making use of these features, our programs make it possible for members of IT support team (i.e. administrators) who are not always familiar with SAS to manage access rights easily. All they have to do is to input the names and roles of involved persons and click an execution button on Excel. In addition, from the viewpoint of program maintenance, SAS seems a more reasonable choice than other software such as Java or etc., since almost all the members are acquainted with SAS in the Biostatistics Department (abbreviated as 'our department' hereafter).

In the anterior half of this paper, we show our network environment and information management process. In the posterior half, we explain how to control access rights via SAS programs with a hypothetical example.

OUR WORK ENVIRONMENT

First, we explain the organization of a project team for drug development in the clinical phase in our company. A project team consists of clinical and analysis teams. Members of our department belong to the analysis team. Analysis team works on SAS server which is managed by our department. Figure 1 shows the organization chart of a project team. The details of the roles shared by the analysis team will be explained later.

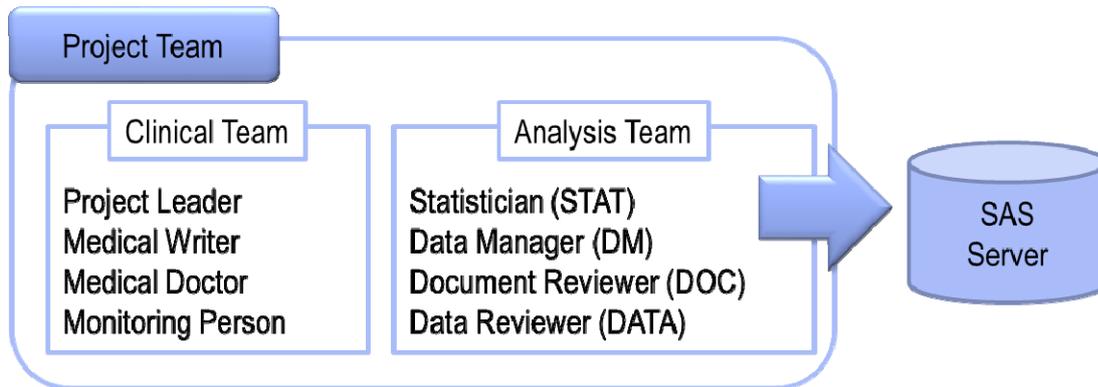


Figure 1. Organization Chart of a Project Team

In our department, we recently integrated a SAS server (used in the Japanese office) with one in the U.S. branch office. Figure 2 shows the framework of our new network environment. For example, the analysis team of 'Project A' consists of members of the Japanese office since 'Project A' develops the drug for Japan. On the other hand, 'Project C' develops the drug globally. Therefore, the analysis team consists of members of the Japanese office and the U.S. branch office. Note that this paper focuses on information management on the SAS server (not SDD).

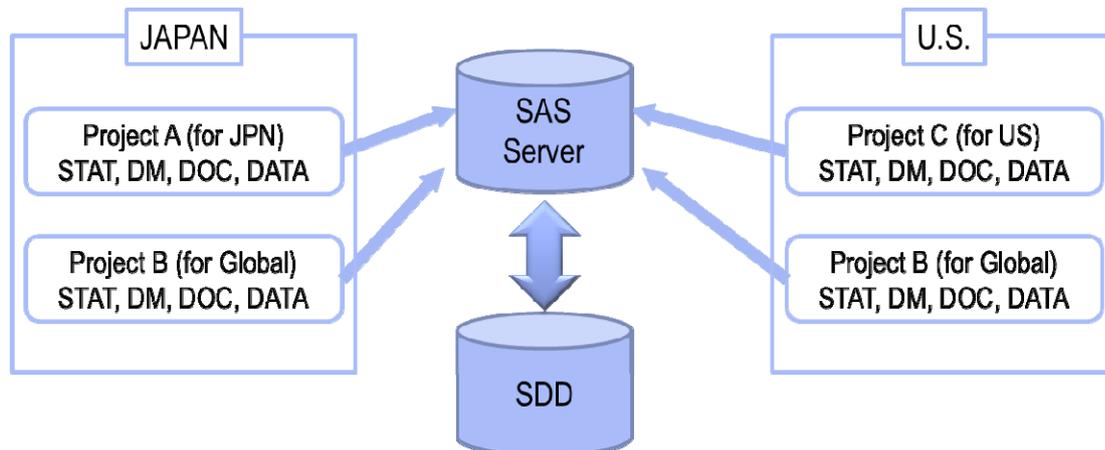


Figure 2. Framework of Our Network Environment

In association with network environment integration, we should have managed information strictly because many persons had become to be able to access the same SAS server. Meanwhile, we made new work folder structure (Figure 3) in reference to Study Data Specifications that FDA had released in 2012 [1]. In addition, we classified analysis team members into four groups based on their roles (i.e. statistician: STAT, data manager: DM, document reviewer: DOC, data reviewer: DATA), where members of STAT, DM, DOC, and DATA constitute a user group. Then, we decided to specify the access rights of respective groups in the analysis team having the same roles and same security rights to respective folders (tied to specific information). Note that STAT comprehends a programmer for statistical analysis and administrator (i.e. ADMIN) is independent of a clinical project.

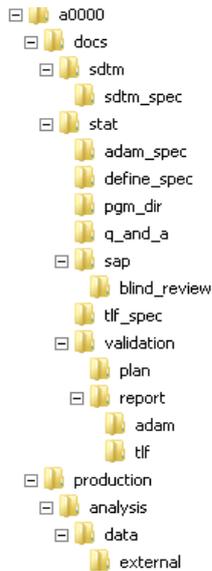


Figure 3. Template of Work Folder Structure (abr.)

DEFINE ACCESS RIGHTS

In our company, the IT department (not our department) has responsibility for information management. In other words, IT persons play the role of administrators. However, IT persons are not always familiar with drug development. Thus, it is difficult for them to determine which role in the analysis team should access what information. For this reason, we prepared 'the Access Rights Sheet' (Figure 4) to define the access rights to each folder. For example (red boxed part in Figure 4), STAT can read, delete and rewrite the files at 'data' folder, and make new files and so on. With regard to DM and DOC, they cannot access this folder because this folder is used for statistical analysis. DATA can only read the files and folders for data (ADaM) review. It should be noted that ADMIN (IT persons) has full control access rights for all folders.

Folder	Remarks	ADMIN (IT person)	STAT	DM	DOC	DATA
project		F	R	R	R	R
protocol		F	R	R	R	R
work		F	C	C	R	R
a0001		F	R	R	R	R
docs		F	R	R	R	R
sdtm		F	R	C	R	R
sdtm_spec	SDTM specification	F	R	C	R	R
stat		F	R	R	R	R
sap	SAP and mock up	F	C	R	R	R
blind_review	Programs and materials for blind review	F	C	R	R	R
adam_spec	ADaM specification	F	C	R	R	R
tlf_spec	TLF specification	F	C	R	R	R
define_spec	define.xls for ADaM and TLF	F	C	R	R	R
pgm_dir	program directory	F	C	R	R	R
q_and_a	question-and-answer list	F	C	R	R	R
validation		F	C	R	R	R
plan	validation plan	F	C	R	R	R
report	validation report	F	C	R	R	R
adam	for ADaM test	F	C	R	R	R
tlf	for TLF test	F	C	R	R	R
production		F	R	L	L	R
analysis		F	R	N	N	R
data	ADaM datasets	F	C	N	N	R
external	External data	F	C	N	N	R
format_e	Format catalog for English	F	C	N	N	R

F: Full control, R: Read Only, C: Can Read and Modify, N: Cannot access, L: Can See Only Lists

Figure 4. Access Rights Sheet (abr.)

INFORMATION MANAGEMENT PROCESS

As you can see, setting access rights in accordance with the 'Access Rights Sheet' (Figure 4) is very tough work because our folder structure is very complicated and access rights are defined minutely. Setting of access rights becomes much tougher if we try to do it manually. Therefore, we developed two SAS programs and named them 'user_management.sas' and 'folder_creation.sas', respectively. In addition, we rearranged the information management process including SAS programs. Figure 5 shows an overview of the information management process.

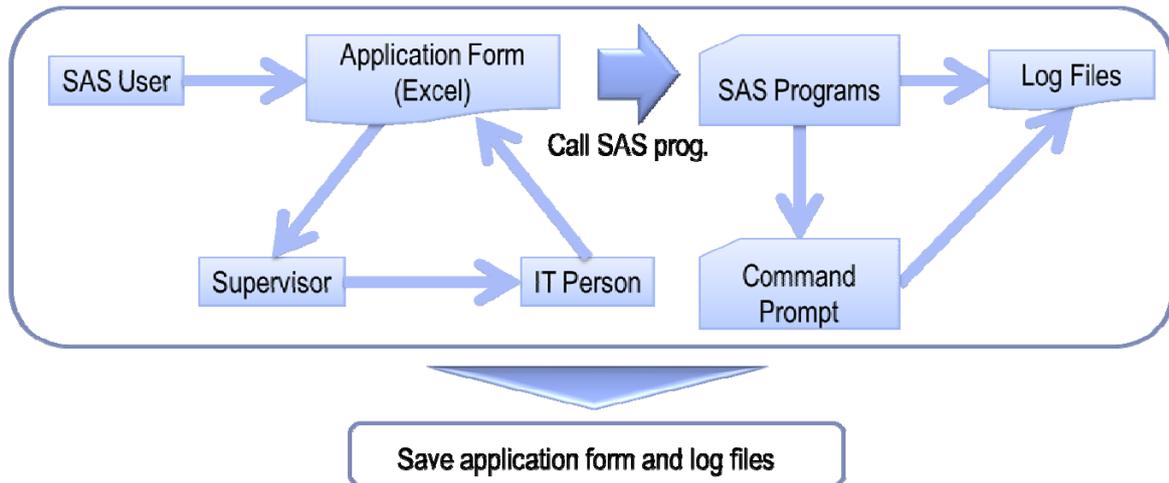


Figure 5. Information Management Process

The point is that all SAS users (e.g. members in our department) have to do is to fill in the application forms (Figure 6 and Figure 7), and send the forms to their supervisor. After the supervisor's approval, the administrator has only to click the 'Execute' button on the form to call user_management.sas or folder_creation.sas. The form and log files are stored to trace the history of access rights.

When a new project starts, members of the analysis team will be assigned. At this time, Figure 6 which is connected to user_management.sas will be submitted. It requires project name, member's role and ID. Namely, user_management.sas assigns members to a specified user group. For example, Figure 6 shows that Yuichi Koretaka is assigned to the s123456 project as STAT, that is, user group (s123456_STAT) is made and assigns Yuichi Koretaka to this user group. The important point to note is that access rights are controlled project-by-project (not protocol-by-protocol with complexity in mind). In addition to user assignment, the project/protocol folder should be created. To create the project/protocol folder, Figure 7 which is connected to folder_creation.sas, will be submitted. It requires project/protocol name and user group name. When the 'Execute' button is clicked, template folder (Figure 3) is copied and renamed project/protocol name via folder_creation.sas with appropriate access rights for the specified user group. For example, Figure 7 shows that new project/protocol (s123456/a0000) starts, that is, project/protocol folders are made and access rights to each folder is set. More details of the SAS programs will be explained later in this paper.

Application Form of User group

To Operation Team	Applicant: Yuichi Koretaka Date: 14FEB2014
-------------------	---

You can insert row if edit box is not enough.

Project Role (User Group)	Name	ID	Add/Delete
s123456_STAT	Yuichi Koretaka	xxxxxx	Add
		#N/A	
		#N/A	
		#N/A	

This button is for administrators only.

Execute

Figure 6. Application Form of User Group

Application Form of Analysis Folder

To Operation Team	Applicant: Yuichi Koretaka	
	Date: 14FEB2014	

Project	s123456	
Protocol	a0000	

Specify User Group

User Group Label

s123456

This button is for administrators only.

Execute

[Execution log](#)

Figure 7. Application Form of Analysis Folder

DETAILS OF TWO SAS PROGRAMS

Two SAS programs (user_management.sas and folder_creation.sas) are included in our information management process. These programs are called through the application form. Namely, when an IT person clicks the 'Execute' button, the connected SAS program is called and outputs the SAS log file by the following VBA code.

```

Sub button()
retval = Shell("[directory]\SAS\SAS9.2\SASFoundation\9.2(32-bit)\sas.exe -config
[directory]\SAS\SAS9.2\SASFoundation\9.2(32-bit)\nls\1d\SASV9.CFG -sysin
[directory]\[program name].sas -log [directory]\[file name].log")
End Sub
    
```

MAKE USER GROUPS AND ASSIGN USERS

As we simply mentioned before, `user_management.sas` is the program for making user groups and assigning selected members to respective user groups when a new project starts. Once the 'Execute' button on the application form (Figure 6) is clicked, `user_management.sas` will be called. Figure 6 shows an example where user group (`s123456_STAT`) is made and Yuichi Koretaka is assigned to `s123456_STAT`. After that, `folder_creation.sas` sets access rights of user groups to each folder as we describe later. Below is a brief workflow of `user_management.sas`. Once the 'Execute' button is clicked by the administrator, all of the processing will be done automatically.

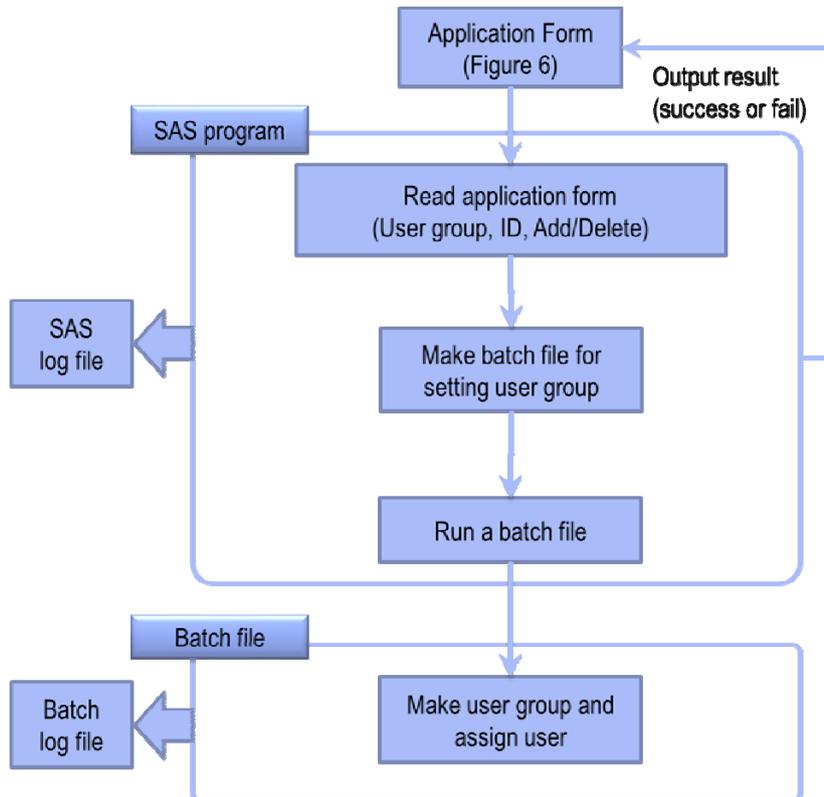


Figure 8. Brief Workflow of `user_management.sas`

This program has no macro parameters. Instead of using macro parameters, it reads the application form (Figure 6: red boxed part) and make SAS data set. Below is a snippet of SAS code. DDE is the old method but it realizes simple communication between SAS and Excel.

```

**** Read Excel data ;
filename EXCEL1 DDE
"EXCEL|[[filename.xls]Application!R10C2:R65536C5" ;
data ___temp01 ;
  length LENGTH LOCAL0 LOCAL NAME NO ADD $100 ;
  infile EXCEL1 missover dsd notab dlm='09'x lrecl=50000;
  input LOCAL0 $ NAME $ NO $ ADD $;
  LOCAL = upcase(LOCAL0);
  if LOCAL^='' and NAME^='' then output;
run ;

```

After that, the program confirms whether the specified user group exists or not. The PIPE engine is useful for confirming the existing user group because the PIPE engine makes it possible for SAS to import the output result of the command prompt. If the specified user group does not exist, the batch file to make user groups will be made. This batch file is run via SAS. If we use the 'x' command, we do not confirm the detailed log of command prompt. Thus, we make a batch file. Namely, we can do accurate error handling through the SAS log and command prompt log. SAS code is shown below.

```
**** Confirm existing user group;
filename data PIPE "net localgroup";
data __temp02;
  infile data dlm='09'x ;
  length TEMP LOCAL $100;
  input TEMP $;
  LOCAL=upcase(compress(TEMP,'*'));
  if substr(TEMP,1,1)='*' and
     LOCAL notin('BACKUP OPERATORS',
                 'DISTRIBUTED COM USERS', 'NETWORK CONFIGURATION OPERATORS',
                 'PERFORMANCE LOG USERS', 'PERFORMANCE MONITOR USERS',
                 'POWER USERS', 'PRINT OPERATORS')
  then output;
  drop TEMP;
run;

proc sql
  create table __temp03 as
  select A.LOCAL
  from __temp01 as A full join __temp02 as B
  on (A.LOCAL=B.LOCAL)
  where (A.LOCAL^='' AND B.LOCAL='');
quit;

data __temp03_2;
  call symput('OBS',put(NOBS,8.0));
  if NOBS=0 then stop;
  set __temp03 NOBS=NOBS end=LASTOBS;
  by LOCAL;
  if first.LOCAL=1 then output;
run ;

%if OBS^=0 %then %do;
  data __temp03_2;
    length COMMAND $200.;
    set __temp03_2;
    COMMAND = "&netlocalgroup. "||trim(LOCAL)||" /add";
  run;

**** Make batch file;
filename txt "Q:\maintenance\standard_macro\g_guser\command.bat";
data _null_;
  set __temp03_2;
  file txt dsd;
  put COMMAND ;
run;
```

As for assigning members to user groups, the SAS code is not shown in this paper but similar code would be enough.

CREATION OF FOLDER AND USER GROUP

As we briefly mentioned before, folder_creation.sas is the program for copying the project and/or protocol folders with appropriate access rights from the template. Once the 'Execute' button on the application form (Figure 7) is clicked, this SAS program will be called. Figure 9 shows a brief workflow of folder_creation.sas.

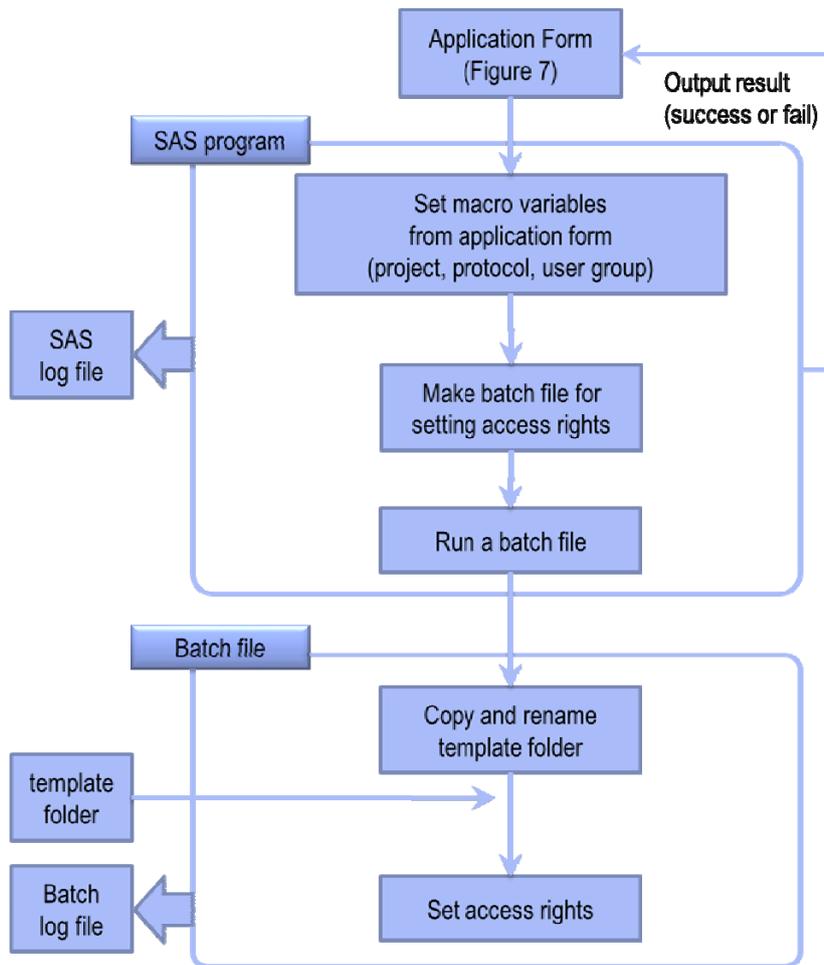


Figure 9. Brief Workflow of folder_creation.sas

This program has no macro parameters, either. Instead of using macro parameters, it reads the application form (Figure 7: red boxed part) and sets macro variables. Then, the batch file is made in accordance with the macro variables. A very simple snippet of example code is shown below.

```

/*****
&project.      : project name      ;
&project.      : protocol name     ;
&user.         : user group label  ;
*****/

filename in DDE "Excel|MAIN|[RANGE]";
data _null_;
  length TEXT $200;
  infile in dlm='09'x notab;
  input TEXT;
  call symput("protocol",TEXT);
run;
%let protocol = %upcase(&protocol.) ;

**** check if the specified project folder exist or not ;
%if( %sysfunc(fileexist(%nrquote([directory]\&project.))) ^= 1 ) %then %do;

```

```

**** execute if the specified project folder does not exist ;
**** copy template folder and set access rights ;
data _temp;
  A1="mkdir [directory]\&project.";
  A2="robocopy [directory]\template [directory]\&project. /E /COPYALL /MIR";
  A3="icacls [directory]\&project. /grant:r
      &user._STAT:(OI)(CI)(RX)
      &user._DM:(OI)(CI)(RX)
      &user._DATA:(OI)(CI)(RX)
      &user._DOC:(OI)(CI)(RX) /T /C";
proc transpose data=_temp out=_temp2(keep=COL1);
  var A1-A3 ;
run;

**** make batch file ;
filename txt "Q:\maintenance\standard_macro\g_gfolder\command.bat" mod;
data _null_;
  set _temp2;
  file txt dsd;
  put COL1 ;
run;

%if( %sysfunc(fileexist(%nrbrquote([directory]\&project.\&protocol.)))^=1 ) %then %do;
  :
  :
  :
%end;

```

This program checks if the specified project and/or protocol folder already exists or not. In case it already exists, an error message will be output to the application form like Figure 10. You may think, this type of error would be detectable from the log file but the administrator may not always understand the SAS contents of the log file. Thus, we implement this function to make it easy for the administrator to understand the error.

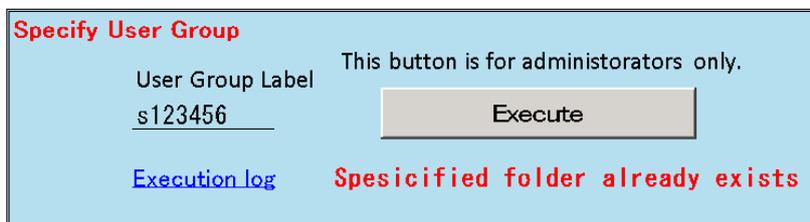


Figure 10. Error Message Example

CONCLUSION

To minimize company risks, information management should be done accurately. Controlling access rights of members depending on their roles in a project is a reasonable way. However, controlling manually is not efficient. To be more efficient and smart, access rights control should be done automatically. As we suggested in this paper, automation by SAS is a very simple way for both administrators and members of analysis team to implement access control. Administrators can control access rights by using our programs even if they are not familiar with clinical development. In addition, we can confirm the detailed log files and trace the history of access rights. It should be emphasized that by using similar techniques shown in this paper, we can change access rights flexibly. We hope this paper will be useful for information management.

REFERENCES

- [1] U.S. Food and Drug Administration, Study Data Specifications Version 2.0, at <http://www.fda.gov/downloads/forindustry/datastandards/studydatastandards/ucm312964.pdf>
- [2] Microsoft®, 2014.02.25, at <http://windows.microsoft.com/en-us/windows/user-groups>

ACKNOWLEDGEMENTS

We would like to thank Hideaki Watanabe and Mitsuaki Machida for supporting us. We would also like to thank the ex-system operation team that made the predecessor of our programs, and all biostatistics members. Special thanks to Wataru Ueda for providing his expertise on DOS programming, and Yoshihide Tsuchiya and Gary E. Moore for correcting my English.

CONTACT INFORMATION

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

Name: Yuichi Koretaka
Company: SHIONOGI & CO., LTD.
Address: Osaka, Japan
Work Phone: +81-6-6485-5094
E-mail: yuichi.koretaka@shionogi.co.jp

Name: Masakazu Fujiwara
Company: SHIONOGI & CO., LTD.
Address: Osaka, Japan
Work Phone: +81-6-6485-5094
E-mail: masakazu.fujiwara@shionogi.co.jp

Name: Yuki Yoshida
Company: SHIONOGI & CO., LTD.
Address: Osaka, Japan
Work Phone: +81-6-6485-5094
E-mail: yuki.yoshida@shionogi.co.jp

Name: Yoshitake Kitanishi
Company: SHIONOGI & CO., LTD.
Address: Osaka, Japan
Work Phone: +81-6-6485-5094
E-mail: yoshitake.kitanishi@shionogi.co.jp

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.