Varied languages, Universal thought: How to handle multilingual data in SAS

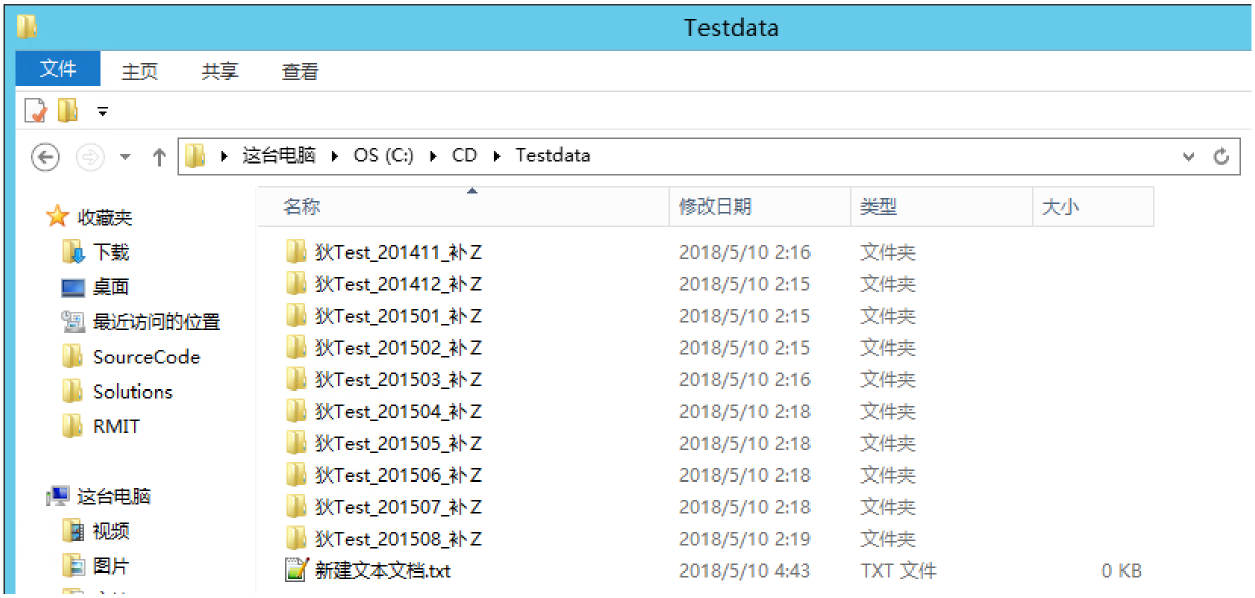
Di Chen, SAS Beijing R&D, Beijing, China

ABSTRACT

Along with the extension and application of CDISC in different countries, more and more non-English data will be processed using SAS. Meanwhile, the changes of data model also affect data preparation, such as add a new table, Add/Drop a column, a dataset added Integrity Constraint, a column occurred Rename, and so on. That causes the data for Extract-Transform-Load (ETL) often require more processing. For impressing and improving the understanding and efficiency, this paper summarize some related knowledges and cautions about preparing, processing and using multi-languages data and present the thought with snippets that how to produce a general process to handle it.

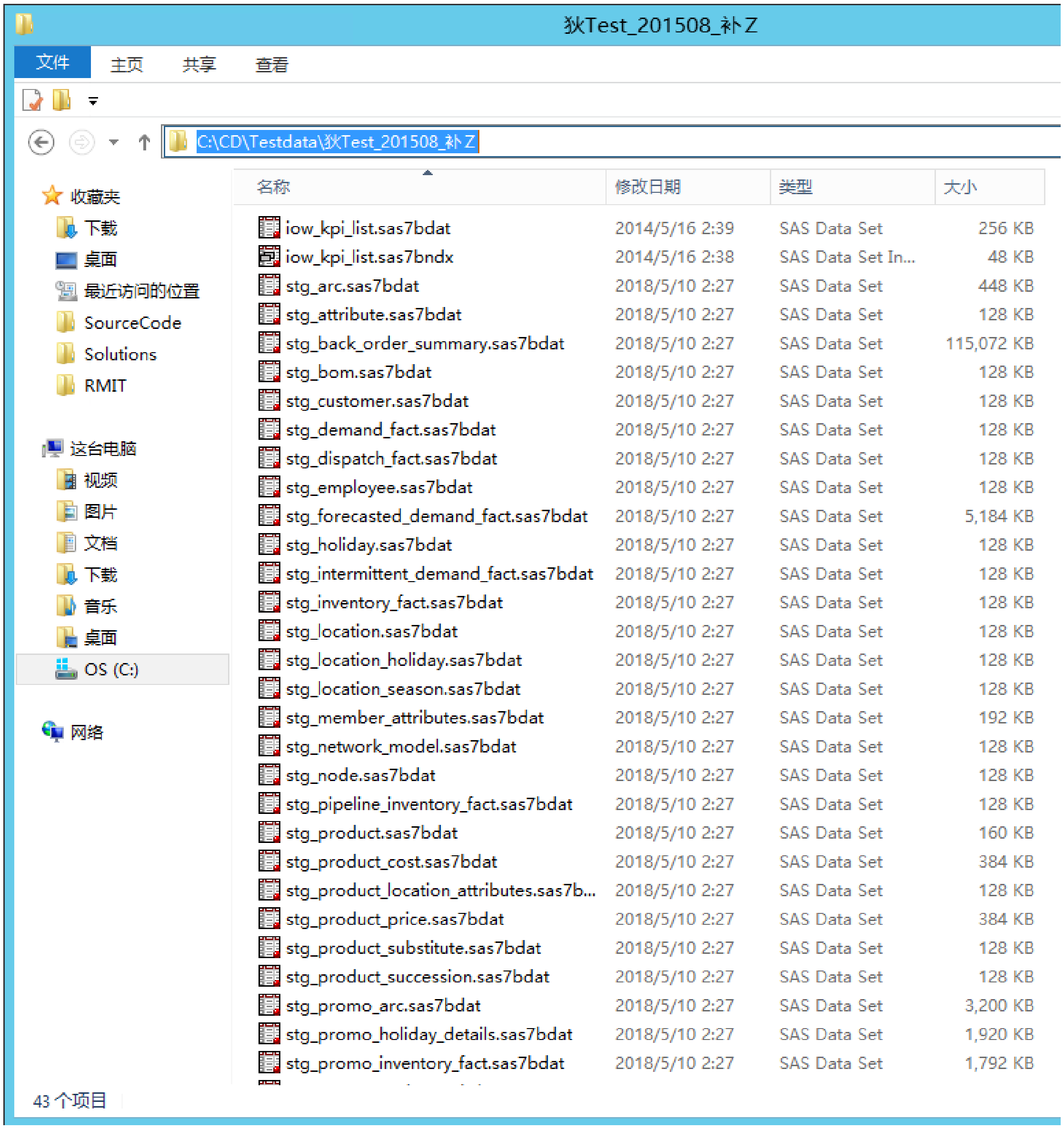
INTRODUCTION

Let's see a simple scenario of most common daily work. I got a zip file from my co-workers which contains folders and data sets. I un-zipped the file into my environment, for example, my environment is WIN with Simplified Chinese. The folders' name contains some DBCS and full-width characters, and the data sets contains international data. Now I want to process the datasets under the folders and updates some values of the datasets.



名称	修改日期	类型	大小
狄Test_201411_补Z	2018/5/10 2:16	文件夹	
狄Test_201412_补Z	2018/5/10 2:15	文件夹	
狄Test_201501_补Z	2018/5/10 2:15	文件夹	
狄Test_201502_补Z	2018/5/10 2:15	文件夹	
狄Test_201503_补Z	2018/5/10 2:16	文件夹	
狄Test_201504_补Z	2018/5/10 2:18	文件夹	
狄Test_201505_补Z	2018/5/10 2:18	文件夹	
狄Test_201506_补Z	2018/5/10 2:18	文件夹	
狄Test_201507_补Z	2018/5/10 2:18	文件夹	
狄Test_201508_补Z	2018/5/10 2:19	文件夹	
新建文本文档.txt	2018/5/10 4:43	TXT 文件	0 KB

Figure 1.1 Libraries



名称	修改日期	类型	大小
iow_kpi_list.sas7bdat	2014/5/16 2:39	SAS Data Set	256 KB
iow_kpi_list.sas7bndx	2014/5/16 2:38	SAS Data Set In...	48 KB
stg_arc.sas7bdat	2018/5/10 2:27	SAS Data Set	448 KB
stg_attribute.sas7bdat	2018/5/10 2:27	SAS Data Set	128 KB
stg_back_order_summary.sas7bdat	2018/5/10 2:27	SAS Data Set	115,072 KB
stg_bom.sas7bdat	2018/5/10 2:27	SAS Data Set	128 KB
stg_customer.sas7bdat	2018/5/10 2:27	SAS Data Set	128 KB
stg_demand_fact.sas7bdat	2018/5/10 2:27	SAS Data Set	128 KB
stg_dispatch_fact.sas7bdat	2018/5/10 2:27	SAS Data Set	128 KB
stg_employee.sas7bdat	2018/5/10 2:27	SAS Data Set	128 KB
stg_forecasted_demand_fact.sas7bdat	2018/5/10 2:27	SAS Data Set	5,184 KB
stg_holiday.sas7bdat	2018/5/10 2:27	SAS Data Set	128 KB
stg_intermittent_demand_fact.sas7bdat	2018/5/10 2:27	SAS Data Set	128 KB
stg_inventory_fact.sas7bdat	2018/5/10 2:27	SAS Data Set	128 KB
stg_location.sas7bdat	2018/5/10 2:27	SAS Data Set	128 KB
stg_location_holiday.sas7bdat	2018/5/10 2:27	SAS Data Set	128 KB
stg_location_season.sas7bdat	2018/5/10 2:27	SAS Data Set	128 KB
stg_member_attributes.sas7bdat	2018/5/10 2:27	SAS Data Set	192 KB
stg_network_model.sas7bdat	2018/5/10 2:27	SAS Data Set	128 KB
stg_node.sas7bdat	2018/5/10 2:27	SAS Data Set	128 KB
stg_pipeline_inventory_fact.sas7bdat	2018/5/10 2:27	SAS Data Set	128 KB
stg_product.sas7bdat	2018/5/10 2:27	SAS Data Set	160 KB
stg_product_cost.sas7bdat	2018/5/10 2:27	SAS Data Set	384 KB
stg_product_location_attributes.sas7b...	2018/5/10 2:27	SAS Data Set	128 KB
stg_product_price.sas7bdat	2018/5/10 2:27	SAS Data Set	384 KB
stg_product_substitute.sas7bdat	2018/5/10 2:27	SAS Data Set	128 KB
stg_product_succession.sas7bdat	2018/5/10 2:27	SAS Data Set	128 KB
stg_promo_arc.sas7bdat	2018/5/10 2:27	SAS Data Set	3,200 KB
stg_promo_holiday_details.sas7bdat	2018/5/10 2:27	SAS Data Set	1,920 KB
stg_promo_inventory_fact.sas7bdat	2018/5/10 2:27	SAS Data Set	1,792 KB

Figure 2.2 Data sets

However, we may fall into several traps during this general scenario as below:

1. Can these folders paths be processed by SAS in my environment successfully?
2. Can these folders' name (the data got from external) be processed by SAS in my environment successfully?
3. Can the international datasets be read or modified correctly?

So especially below points need special attention when dealing with multilingual data:

1. The encoding of SAS session.
2. Transcoding for the data got from external.
3. Transcoding for SAS data sets.

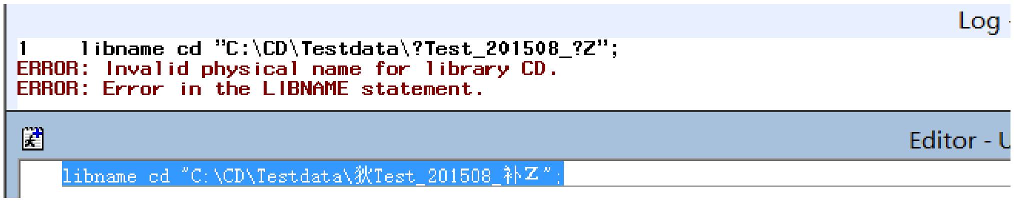
THE ENCODING OF SAS SESSION

As we known, encoding is ubiquitous in SAS 9.

The SAS session encoding establishes the environment to process SAS syntax and SAS data sets, and to read and write external files. We must pay attention to it when operate the international data.

As mentioned in introduction section, there are **three problems** need our attention when processing data under all the folders using SAS:

Problem 1: Can these folders paths be processed by SAS on my environment successfully?



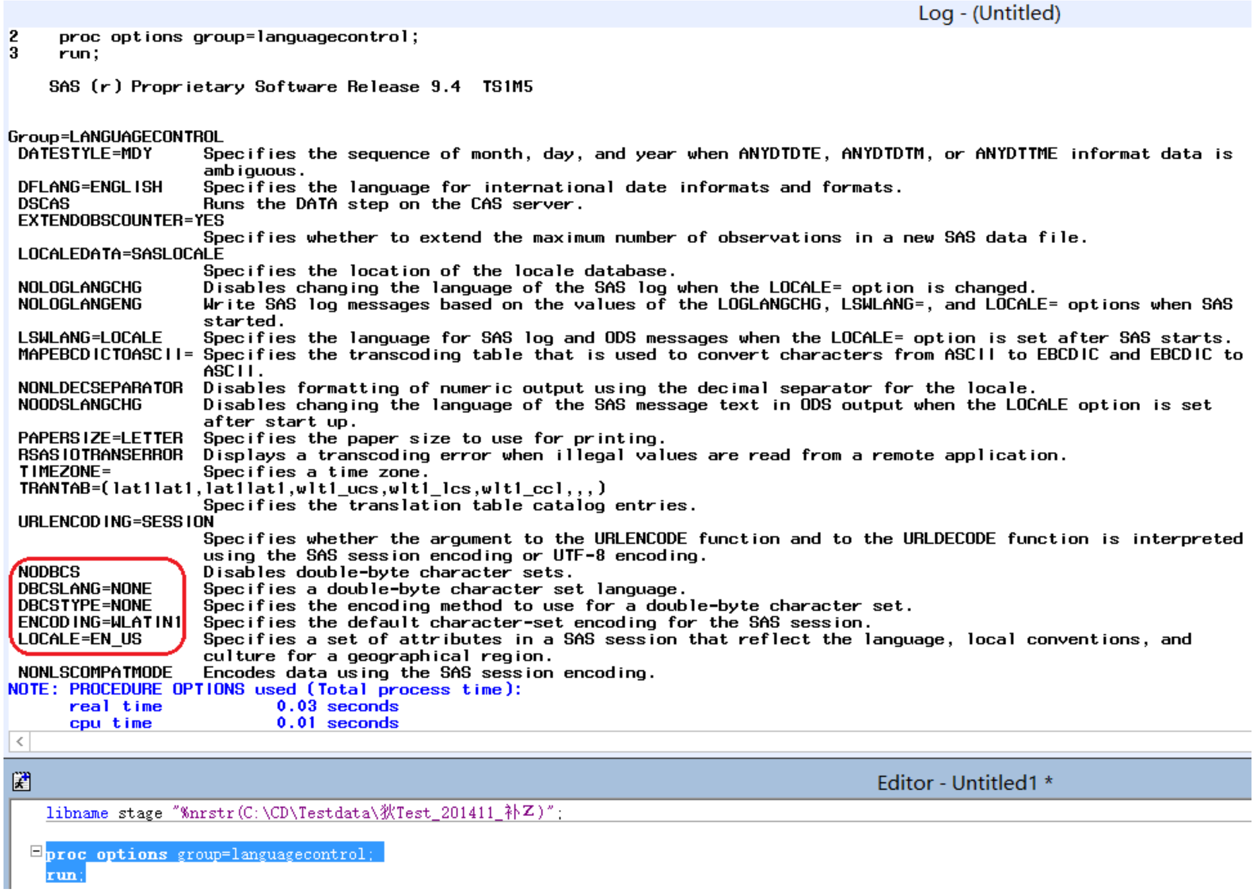
```
1     libname cd "C:\CD\Testdata\?Test_201508_?Z";
ERROR: Invalid physical name for library CD.
ERROR: Error in the LIBNAME statement.
```

```
libname cd "C:\CD\Testdata\狄Test_201508_补Z";
```

Figure 2.1 Error occurs when libname statement run

Do you know why we meet this kind of issues? Why the DBCS characters are changed to “?” ? Let’s see the backend process.

We can use options procedure to get the information of all the options value regarding encoding and locale in current SAS session.



```
2     proc options group=languagecontrol;
3     run;

    SAS (r) Proprietary Software Release 9.4  TS1M5


Group=LANGUAGECONTROL
 DATESTYLE=MDY      Specifies the sequence of month, day, and year when ANYDTDTE, ANYDTDTM, or ANYDTTME informat data is
                    ambiguous.
 DFLANG=ENGLISH     Specifies the language for international date informats and formats.
 DSCAS              Runs the DATA step on the CAS server.
 EXTENDOBSCOUNTER=YES
                    Specifies whether to extend the maximum number of observations in a new SAS data file.
 LOCALEDATA=SASLOCALE
                    Specifies the location of the locale database.
 NOLOGLANGCHG       Disables changing the language of the SAS log when the LOCALE= option is changed.
 NOLOGLANGENG       Write SAS log messages based on the values of the LOGLANGCHG, LSWLANG=, and LOCALE= options when SAS
                    started.
 LSWLANG=LOCALE     Specifies the language for SAS log and ODS messages when the LOCALE= option is set after SAS starts.
 MAPEBCDICTOASCII=  Specifies the transcoding table that is used to convert characters from ASCII to EBCDIC and EBCDIC to
                    ASCII.
 NONLDECSEPARATOR   Disables formatting of numeric output using the decimal separator for the locale.
 NOODSLANGCHG       Disables changing the language of the SAS message text in ODS output when the LOCALE option is set
                    after start up.
 PAPERSIZE=LETTER   Specifies the paper size to use for printing.
 RSASIOTRANSERROR   Displays a transcoding error when illegal values are read from a remote application.
 TIMEZONE=          Specifies a time zone.
 TRANTAB=(lat1lat1,lat1lat1,wlt1_ucs,wlt1_lcs,wlt1_ccl,,,)
                    Specifies the translation table catalog entries.
 URLENCODING=SESSION
                    Specifies whether the argument to the URLENCODE function and to the URLDECODE function is interpreted
                    using the SAS session encoding or UTF-8 encoding.
 NODBCS             Disables double-byte character sets.
 DBCSLANG=NONE      Specifies a double-byte character set language.
 DBCSTYPE=NONE      Specifies the encoding method to use for a double-byte character set.
 ENCODING=WLATIN1   Specifies the default character-set encoding for the SAS session.
 LOCALE=EN_US       Specifies a set of attributes in a SAS session that reflect the language, local conventions, and
                    culture for a geographical region.
 NONLSCOMPATMODE    Encodes data using the SAS session encoding.
NOTE: PROCEDURE OPTIONS used (Total process time):
      real time           0.03 seconds
      cpu time            0.01 seconds
```

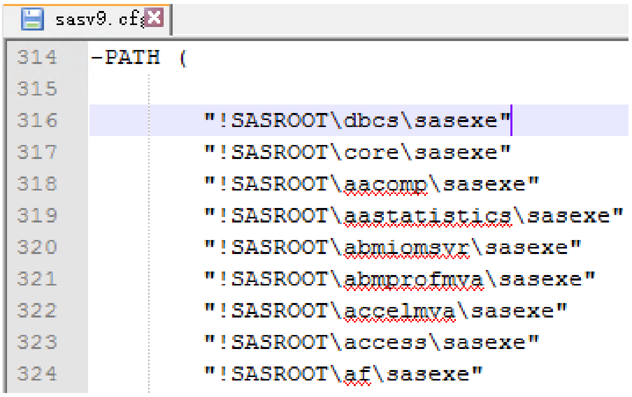
```
libname stage "%nrstr(C:\CD\Testdata\狄Test_201411_补Z)";

proc options group=languagecontrol;
run;
```

Figure 2.2 languagecontrol

3 options can determine or affect SAS session encoding: ENCODING=, LOCALE= and DBCS.

- The ENCODING= system option is used to specify the SAS session encoding. It regardless of whether the DBCS or LOCALE= options are specified. If the ENCODING= option is specified, a set of valid DBCS options is set regardless of whether the user has specified those options. That means the ENCODING= system option has priority. Also, if the ENCODING= option is specified, the LOCALE= option is set to an appropriate value unless a value has been specified by the user. The ENCODING system option is set explicitly in all SAS Foundation sasv9.cfg configuration files.

- Meanwhile, the LOCALE= system option implicitly sets the ENCODING= option if the ENCODING= option is not set explicitly. When install SAS foundation, the default locale depending on the OS locale. Also, you can collect which locales to install.
- The DBCS option is valid only when the DBCS extension directory is included in the path option list. The path of the DBCS extension dynamic link library (DLLs) must be located at the top of the pathname list of the path option for the DBCS languages when you want to invoke a DBCS SAS session. The DBCS extension DLLs are located in the directory !SASROOT/dbcs/sasexe by default.

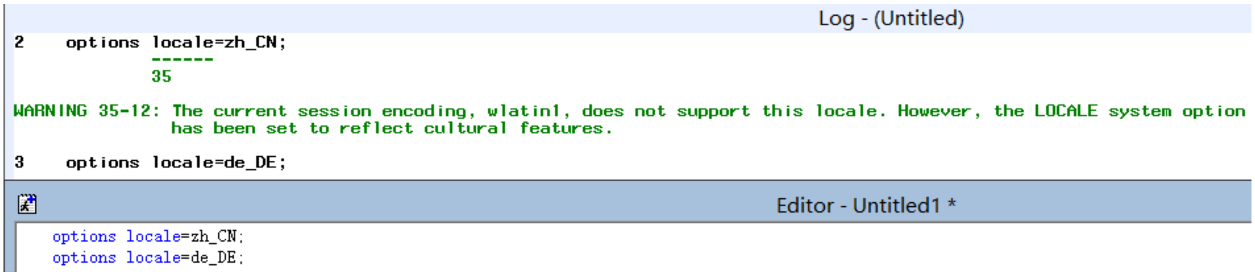


```
-PATH (

        "!SASROOT\dbcs\sasexe"
        "!SASROOT\core\sasexe"
        "!SASROOT\aacomp\sasexe"
        "!SASROOT\aastatistics\sasexe"
        "!SASROOT\abmiomsvr\sasexe"
        "!SASROOT\abmprofmva\sasexe"
        "!SASROOT\accelmva\sasexe"
        "!SASROOT\access\sasexe"
        "!SASROOT\af\sasexe"
```

Figure 2.3 DBCS DLLs

However, the ENCODING= system option is the recommended method in setting a SAS session for DBCS because it takes effect at a SAS session startup.

- You can't use OPTIONS statement in your SAS code to change ENCODING= or DBCS.
- You can use OPTIONS statement in your SAS code to change locale. **Note:** You will get warning when change current locale to a locale which is not supported by current SAS session encoding. For example, under SAS session encoding wlatin1, change locale from en_US to zh_CN will be warned, and de_DE successfully.



```
2    options locale=zh_CN;
                ------
                35

WARNING 35-12: The current session encoding, wlatin1, does not support this locale. However, the LOCALE system option
               has been set to reflect cultural features.

3    options locale=de_DE;
```

```
options locale=zh_CN;
options locale=de_DE;
```

Figure 2.4 Options change locale

The priority order for setting the encoding is as follows:

- ENCODING= system option
- DBCS options
- LOCALE= system option

Generally, we launch SAS foundation from START menu.

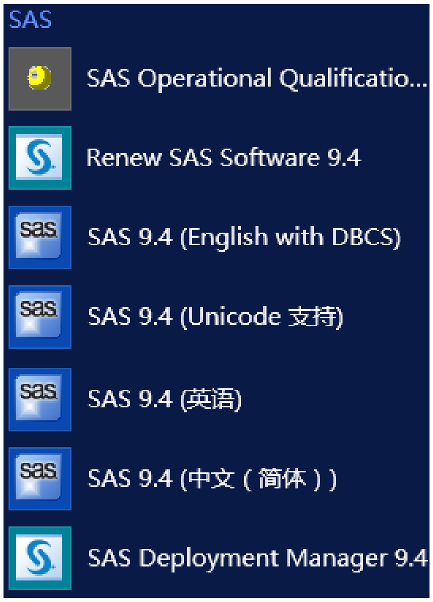
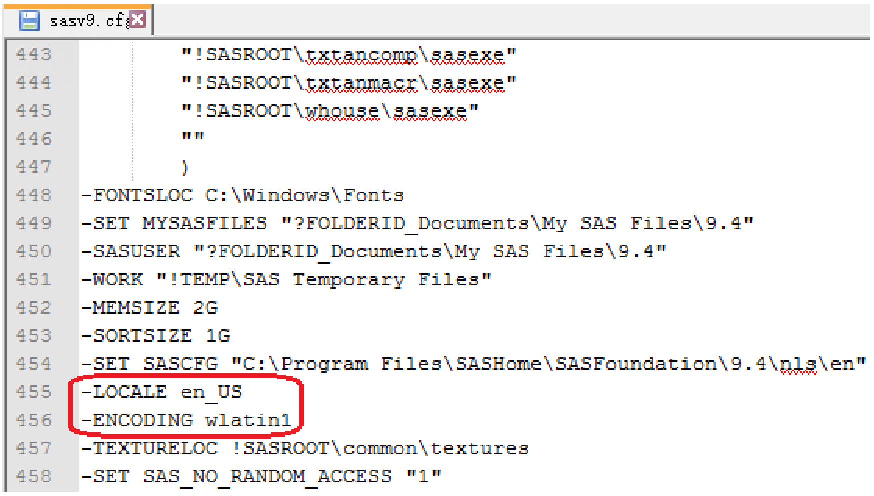


Figure 2.5 Launch SAS Foundation

In fact, the command of launching SAS foundation is as below:

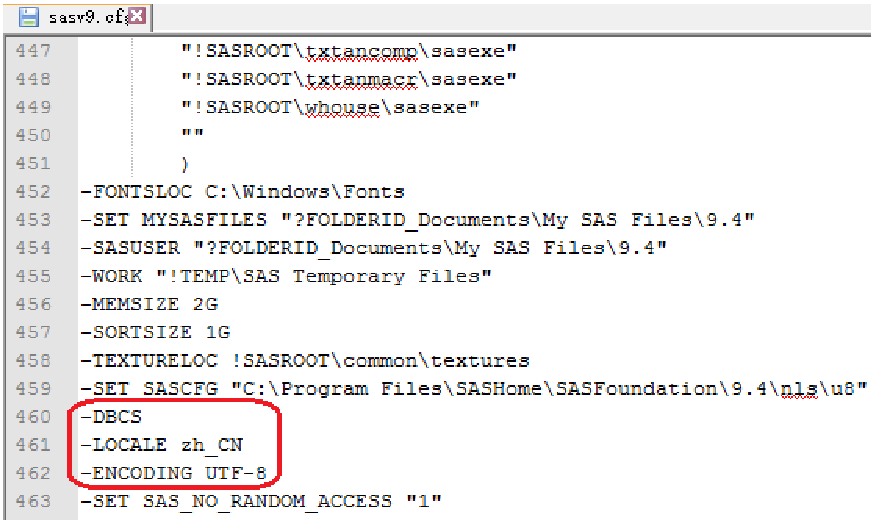
"C:\Program Files\SASHome\SASFoundation\9.4\sas.exe" -CONFIG "C:\Program Files\SASHome\SASFoundation\9.4\nls\en\sasv9.cfg"

The options which are invoked to determine or affect SAS session encoding are set in each sasv9 configure file as below:



```
        "!SASROOT\txtancomp\sasexe"
        "!SASROOT\txtanmacr\sasexe"
        "!SASROOT\whouse\sasexe"
        ""
        )
-FONTSLOC C:\Windows\Fonts
-SET MYSASFILES "?FOLDERID_Documents\My SAS Files\9.4"
-SASUSER "?FOLDERID_Documents\My SAS Files\9.4"
-WORK "!TEMP\SAS Temporary Files"
-MEMSIZE 2G
-SORTSIZE 1G
-SET SASCFG "C:\Program Files\SASHome\SASFoundation\9.4\nls\en"
-LOCALE en_US
-ENCODING wlatin1
-TEXTURELOC !SASROOT\common\textures
-SET SAS_NO_RANDOM_ACCESS "1"
```

Figure 2.6 Options in en configure



```
        "!SASROOT\txtancomp\sasexe"
        "!SASROOT\txtanmacr\sasexe"
        "!SASROOT\whouse\sasexe"
        ""
        )
-FONTSLOC C:\Windows\Fonts
-SET MYSASFILES "?FOLDERID_Documents\My SAS Files\9.4"
-SASUSER "?FOLDERID_Documents\My SAS Files\9.4"
-WORK "!TEMP\SAS Temporary Files"
-MEMSIZE 2G
-SORTSIZE 1G
-TEXTURELOC !SASROOT\common\textures
-SET SASCFG "C:\Program Files\SASHome\SASFoundation\9.4\nls\u8"
-DBCS
-LOCALE zh_CN
-ENCODING UTF-8
-SET SAS_NO_RANDOM_ACCESS "1"
```

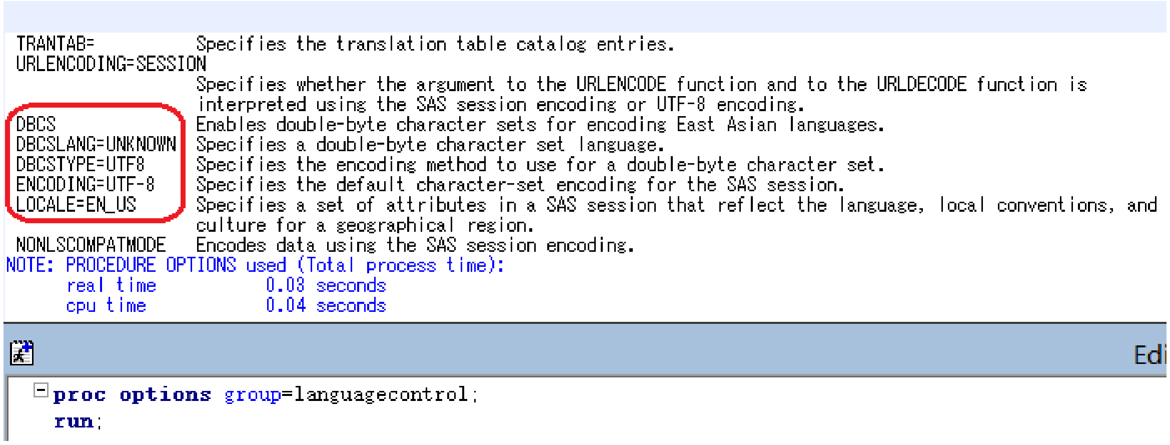
Figure 2.7 Options in u8 configure

We all know the unicode is a multi-byte character set that was created to support all languages. It includes all characters from all modern written language. SAS supports the unicode character set with a session encoding of UTF-8.

We can make a small experiment to prove the importance of the ENCODING system option: In the sasv9 of u8, leave ENCODING=UTF-8 system option and remove DBCS and LOCALE. Then start a SAS session using below command:

"C:\Program Files\SASHome\SASFoundation\9.4\sas.exe" -CONFIG "C:\Program Files\SASHome\SASFoundation\9.4\nls\u8\sasv9.cfg"

Now the options values are as below:

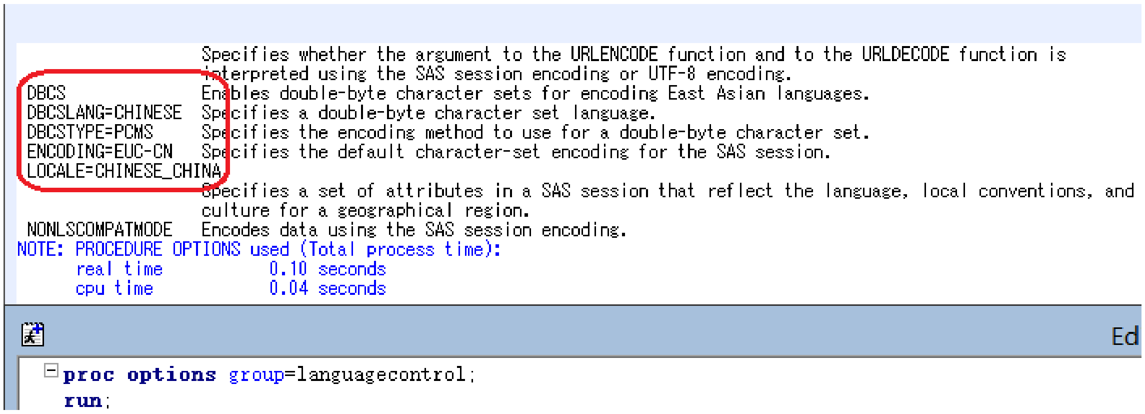


```
 TRANTAB=          Specifies the translation table catalog entries.
 URLENCODING=SESSION
                   Specifies whether the argument to the URLENCODE function and to the URLDECODE function is
                   interpreted using the SAS session encoding or UTF-8 encoding.
 DBCS              Enables double-byte character sets for encoding East Asian languages.
 DBCSLANG=UNKNOWN  Specifies a double-byte character set language.
 DBCSTYPE=UTF8     Specifies the encoding method to use for a double-byte character set.
 ENCODING=UTF-8    Specifies the default character-set encoding for the SAS session.
 LOCALE=EN_US      Specifies a set of attributes in a SAS session that reflect the language, local conventions, and
                   culture for a geographical region.
 NONLSCOMPATMODE   Encodes data using the SAS session encoding.
NOTE: PROCEDURE OPTIONS used (Total process time):
      real time           0.03 seconds
      cpu time            0.04 seconds
```

```
proc options group=languagecontrol;
run;
```

Figure 2.8 Experiment for ENCODING=UTF-8

Then update ENCODING=euc-cn and start another SAS session using the same command.



```
                    Specifies whether the argument to the URLENCODE function and to the URLDECODE function is
                    interpreted using the SAS session encoding or UTF-8 encoding.
 DBCS               Enables double-byte character sets for encoding East Asian languages.
 DBCSLANG=CHINESE   Specifies a double-byte character set language.
 DBCSTYPE=PCMS      Specifies the encoding method to use for a double-byte character set.
 ENCODING=EUC-CN    Specifies the default character-set encoding for the SAS session.
 LOCALE=CHINESE_CHINA
                    Specifies a set of attributes in a SAS session that reflect the language, local conventions, and
                    culture for a geographical region.
 NONLSCOMPATMODE    Encodes data using the SAS session encoding.
NOTE: PROCEDURE OPTIONS used (Total process time):
      real time           0.10 seconds
      cpu time            0.04 seconds
```

```
proc options group=languagecontrol;
  run;
```

Figure 2.9 Experiment: ENCODING=EUC-CN

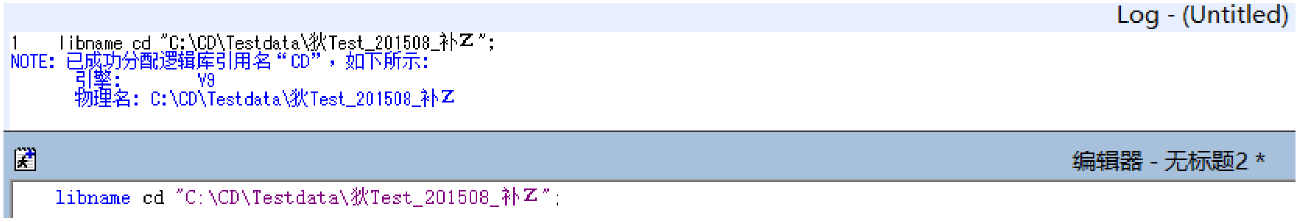
As mentioned above, “If the ENCODING= option is specified, a set of valid DBCS options is set regardless of whether the user has specified those options. And the LOCALE= option is set to an appropriate value unless a value has been specified by the user.” The experiment illustrates that DBCS option does not need set explicitly when encoding is set to non-SBCS encoding.

Additional, we notice another 2 options in the output of proc options: DBCSLANG and DBCSTYPE.

- DBCSLANG: Specifies a double-byte character set language.
- DBCSTYPE: Specifies the encoding method to use for a double-byte character set.

Beginning with SAS 9.3, DBCSLANG= and DBCSTYPE= are not set explicitly. For example, LOCALE=euc-cn will set DBCSLANG=CHINESE (simplified) and DBCSTYPE=EUC; LOCALE=ja_JP will set DBCSLANG=JAPANESE and DBCSTYPE=PCMS.

Now we use ENCODING=UTF-8 that sets DBCS option valid to launch a SAS session. The libref can be created successfully.



```
1    libname cd "C:\CD\Testdata\狄Test_201508_补Z";
NOTE: 已成功分配逻辑库引用名“CD”，如下所示:
      引擎:        V9
      物理名: C:\CD\Testdata\狄Test_201508_补Z
```

```
libname cd "C:\CD\Testdata\狄Test_201508_补Z";
```

Figure 2.10 Libname statement succeed

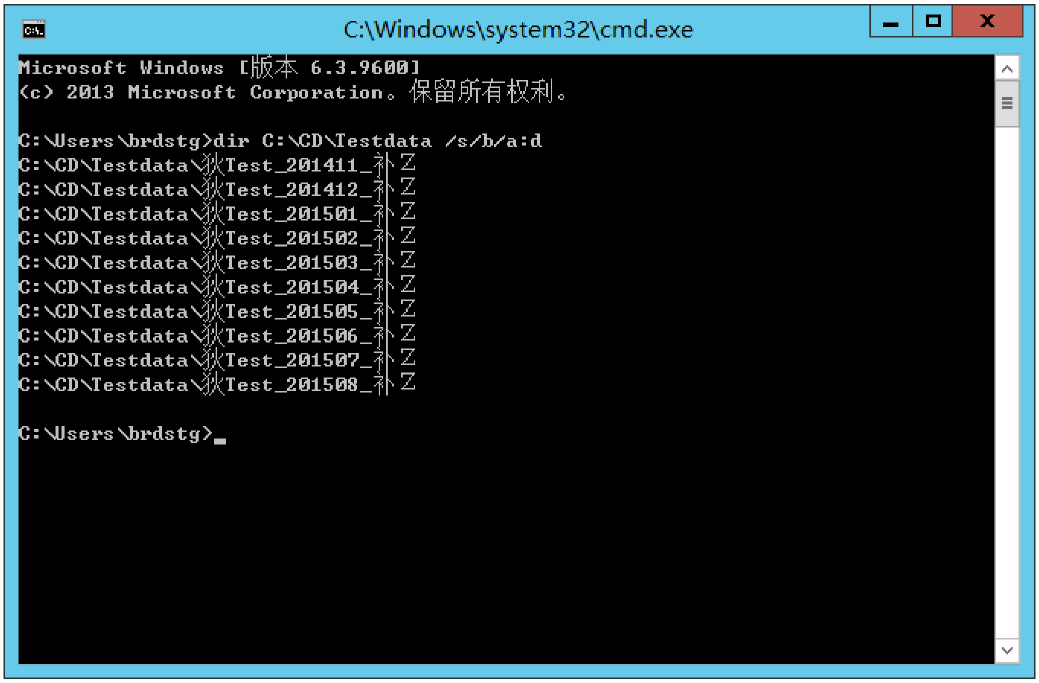
Conclusion: UTF-8 is the safest SAS session encoding for non-English data.

TRANSCODING FOR THE DATA GOT FROM EXTERNAL

Tips: The efficiency of creating libref

Will we change the path value every time for creating libref for each library?

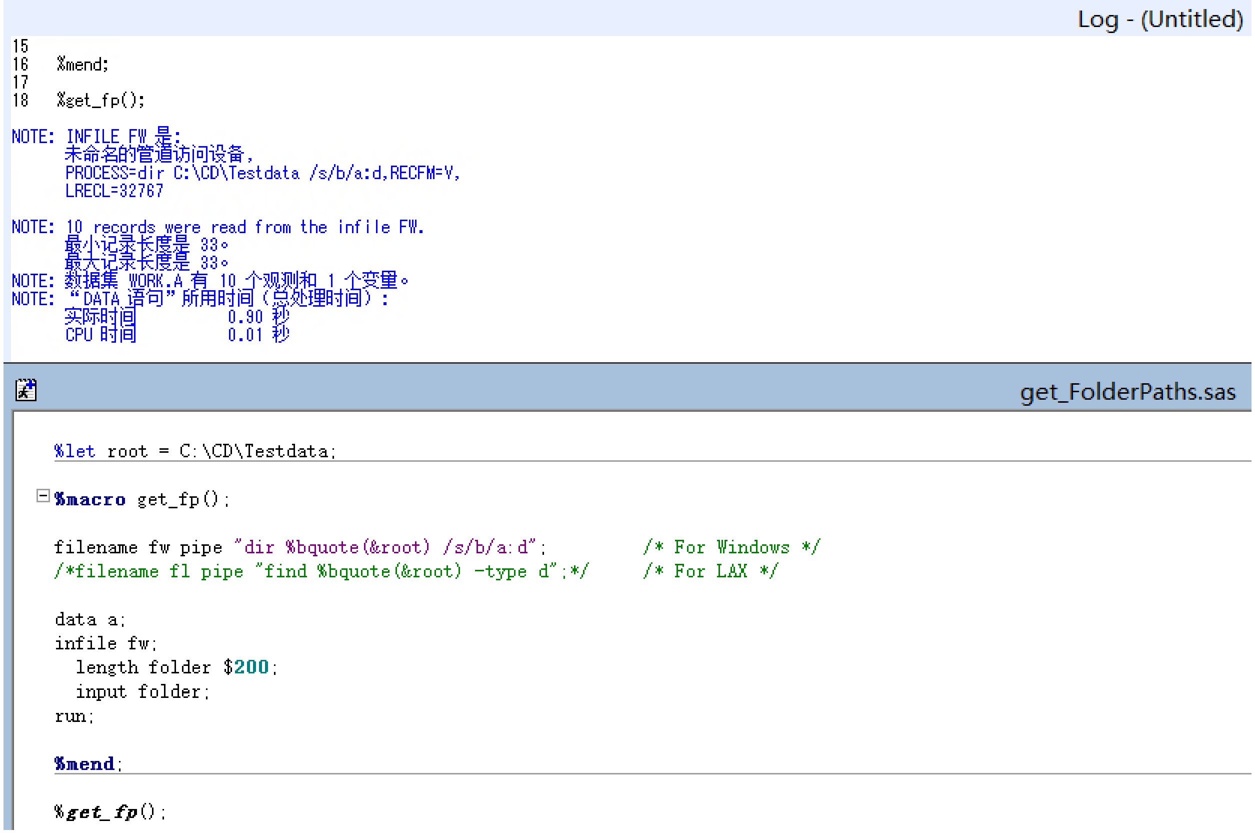
We can get all the paths using OS command to raise the efficiency, i.e. dir for Windows and find for LAX.



```
Microsoft Windows [版本 6.3.9600]
(c) 2013 Microsoft Corporation。保留所有权利。

C:\Users\brdstg>dir C:\CD\Testdata /s/b/a:d
C:\CD\Testdata\狄Test_201411_补Z
C:\CD\Testdata\狄Test_201412_补Z
C:\CD\Testdata\狄Test_201501_补Z
C:\CD\Testdata\狄Test_201502_补Z
C:\CD\Testdata\狄Test_201503_补Z
C:\CD\Testdata\狄Test_201504_补Z
C:\CD\Testdata\狄Test_201505_补Z
C:\CD\Testdata\狄Test_201506_补Z
C:\CD\Testdata\狄Test_201507_补Z
C:\CD\Testdata\狄Test_201508_补Z

C:\Users\brdstg>_
```

Figure 3.1 Command - get all the libraries path in Windows

Then we can use unnamed pipe to generate a data set which contains all the libraries path.



```
15
16    %mend;
17
18    %get_fp();

NOTE: INFILE FW 是:
      未命名的管道访问设备,
      PROCESS=dir C:\CD\Testdata /s/b/a:d,RECFM=V,
      LRECL=32767

NOTE: 10 records were read from the infile FW.
      最小记录长度是 33。
      最大记录长度是 33。
NOTE: 数据集 WORK.A 有 10 个观测和 1 个变量。
NOTE: “DATA 语句”所用时间（总处理时间）:
      实际时间          0.90 秒
      CPU 时间          0.01 秒
```

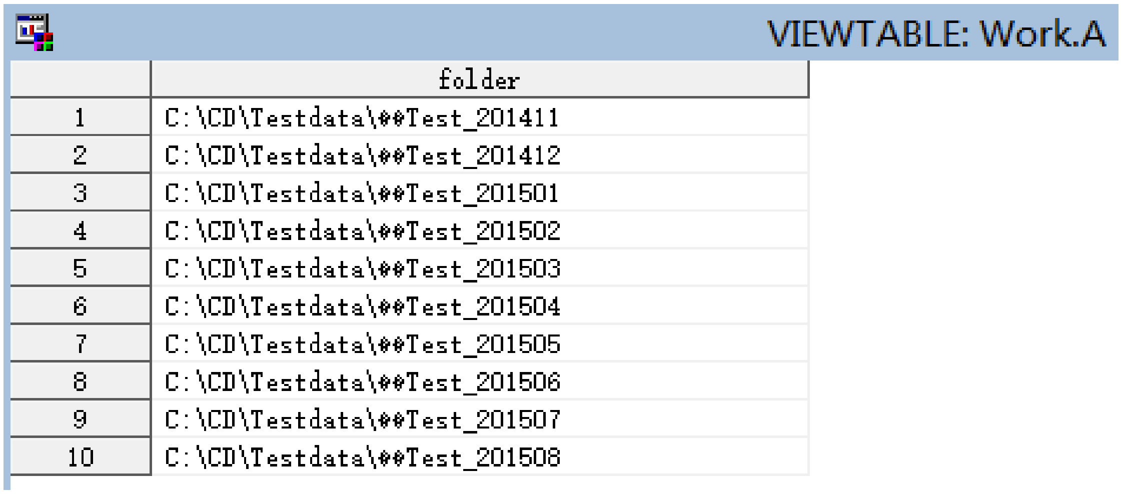
```
%let root = C:\CD\Testdata;

%macro get_fp();

filename fw pipe "dir %bquote(&root) /s/b/a:d";        /* For Windows */
/*filename fl pipe "find %bquote(&root) -type d";*/     /* For LAX */

data a;
infile fw;
  length folder $200;
  input folder;
run;

%mend;

%get_fp();
```

Figure 3.2 Unnamed pipe statement

Unnamed pipes enable you to invoke a program outside of SAS and redirect the program's input, output, and error messages to SAS fileref. This capability enables you to capture data from a program external to SAS without creating an intermediate data file. To use an unnamed pipe, issue a FILENAME statement. That is to say, the unnamed pipe is an extension of the FILENAME statement.

Problem 2: Can these folders' name (the data got from external) be processed by SAS in my environment successfully?

Open or print the dataset a. Then we get garbage code substitute for non-English characters in the paths.



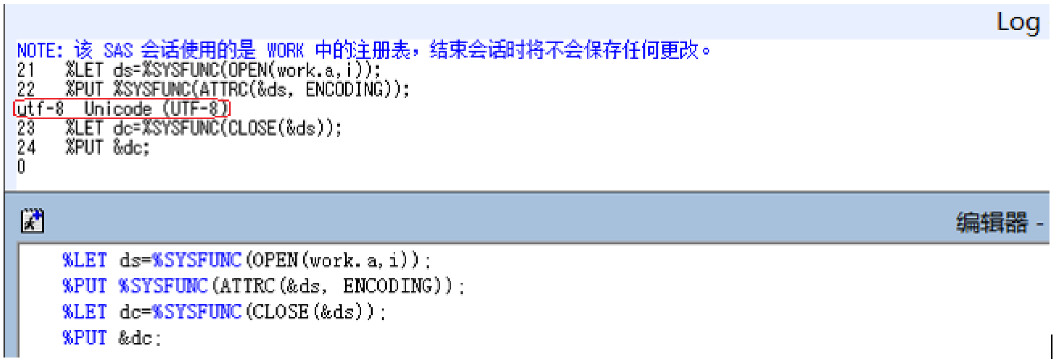
VIEWTABLE: Work.A

	folder
1	C:\CD\Testdata\��Test_201411
2	C:\CD\Testdata\��Test_201412
3	C:\CD\Testdata\��Test_201501
4	C:\CD\Testdata\��Test_201502
5	C:\CD\Testdata\��Test_201503
6	C:\CD\Testdata\��Test_201504
7	C:\CD\Testdata\��Test_201505
8	C:\CD\Testdata\��Test_201506
9	C:\CD\Testdata\��Test_201507
10	C:\CD\Testdata\��Test_201508

Figure 3.3 Garbage in dataset

What's the matter and how to fix it?

We can use below code to get the encoding of dataset a.



```
NOTE: 该 SAS 会话使用的是 WORK 中的注册表，结束会话时将不会保存任何更改。
21   %LET ds=%SYSFUNC(OPEN(work.a,i));
22   %PUT %SYSFUNC(ATTRC(&ds, ENCODING));
utf-8  Unicode (UTF-8)
23   %LET dc=%SYSFUNC(CLOSE(&ds));
24   %PUT &dc;
0
```

```
    %LET ds=%SYSFUNC(OPEN(work.a,i));
    %PUT %SYSFUNC(ATTRC(&ds, ENCODING));
    %LET dc=%SYSFUNC(CLOSE(&ds));
    %PUT &dc;
```

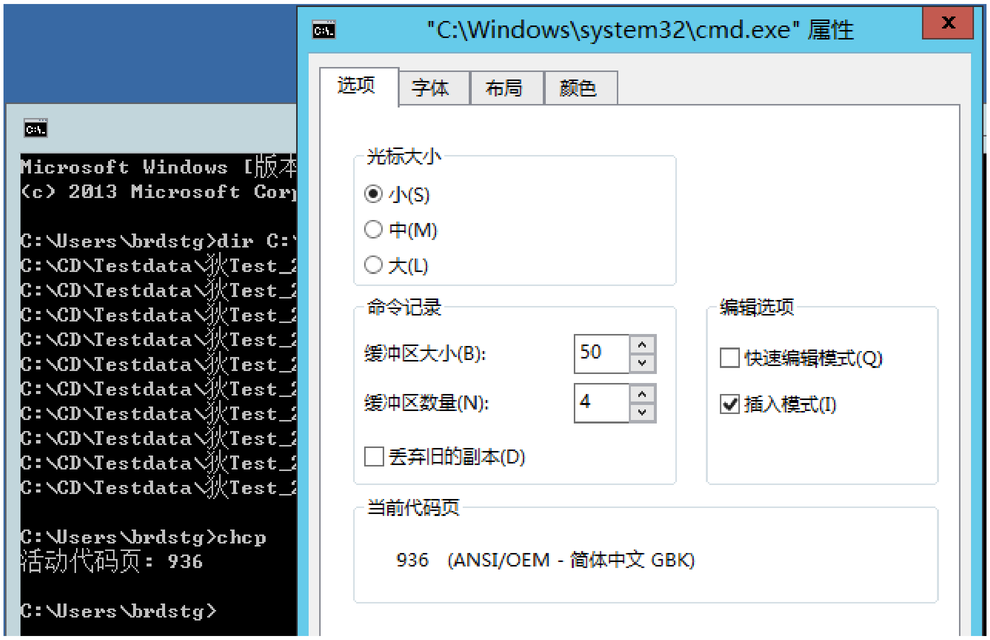
Figure 3.4 Encoding of dataset

Why the non-English characters in dataset with utf-8 encoding decodes failed in SAS session with UTF-8 encoding, although they are processed successfully in libname statement?

We know the paths are got from OS through unnamed pipe that means the encoding of the external data is determined by OS. **So the reason of garbage code is, the data which encoded using OS encoding is decoded using SAS session's encoding UTF-8**.

Transcoding is the process of converting data from one encoding to another. Transcoding is necessary when the SAS session encoding and the encoding of the data are different. We must transcode them to UTF-8 once they stored in dataset.

What is the OS encoding? We can get it using chcp command or from properties of cmd window.



```
Microsoft Windows [版本
(c) 2013 Microsoft Cor

C:\Users\brdstg>dir C:
C:\CD\Testdata\狄Test_2
C:\CD\Testdata\狄Test_2
C:\CD\Testdata\狄Test_2
C:\CD\Testdata\狄Test_2
C:\CD\Testdata\狄Test_2
C:\CD\Testdata\狄Test_2
C:\CD\Testdata\狄Test_2
C:\CD\Testdata\狄Test_2
C:\CD\Testdata\狄Test_2
C:\CD\Testdata\狄Test_2

C:\Users\brdstg>chcp
活动代码页: 936

C:\Users\brdstg>
```

"C:\Windows\system32\cmd.exe" 属性

光标大小

● 小(S)
○ 中(M)
○ 大(L)

命令记录

缓冲区大小(B): 50

缓冲区数量(N): 4

□ 丢弃旧的副本(D)

编辑选项

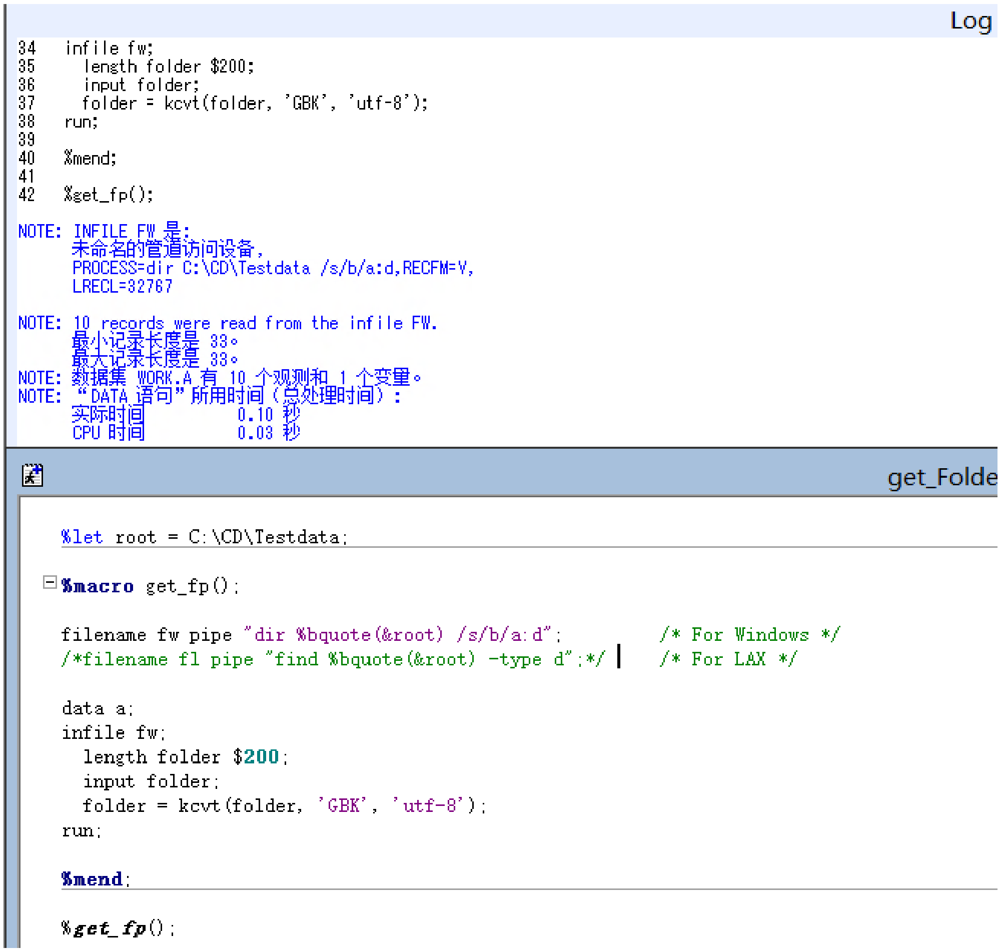
□ 快速编辑模式(Q)
☑ 插入模式(I)

当前代码页

936 (ANSI/OEM - 简体中文 GBK)

Figure 3.5 Encoding of OS (Simplified Chinese Windows)

Now we can use function KCVT to do the transcoding for the data got from Windows.

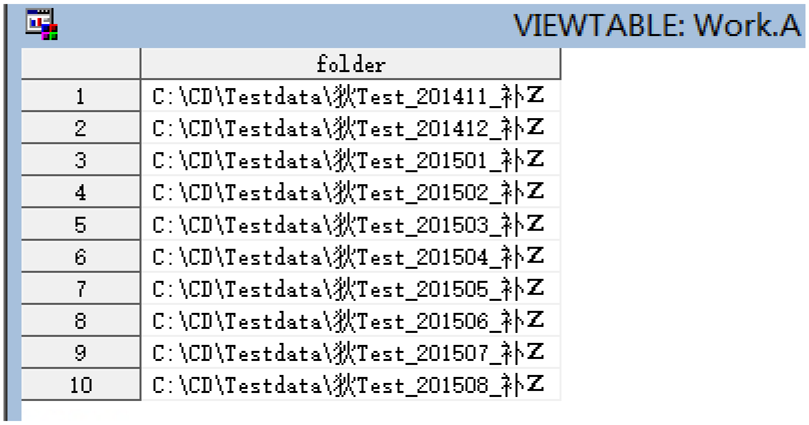


```
   infile fw;
     length folder $200;
     input folder;
     folder = kcvt(folder, 'GBK', 'utf-8');
   run;

   %mend;

   %get_fp();

NOTE: INFILE FW 是:
      未命名的管道访问设备,
      PROCESS=dir C:\CD\Testdata /s/b/a:d,RECFM=V,
      LRECL=32767

NOTE: 10 records were read from the infile FW.
      最小记录长度是 33。
      最大记录长度是 33。
NOTE: 数据集 WORK.A 有 10 个观测和 1 个变量。
NOTE: “DATA 语句”所用时间(总处理时间):
      实际时间            0.10 秒
      CPU 时间            0.03 秒
```

```
%let root = C:\CD\Testdata;

%macro get_fp();

filename fw pipe "dir %bquote(&root) /s/b/a:d";          /* For Windows */
/*filename fl pipe "find %bquote(&root) -type d";*/     /* For LAX */

data a;
infile fw;
  length folder $200;
  input folder;
  folder = kcvt(folder, 'GBK', 'utf-8');
run;

%mend;

%get_fp();
```

Figure 3.6 KCVT function

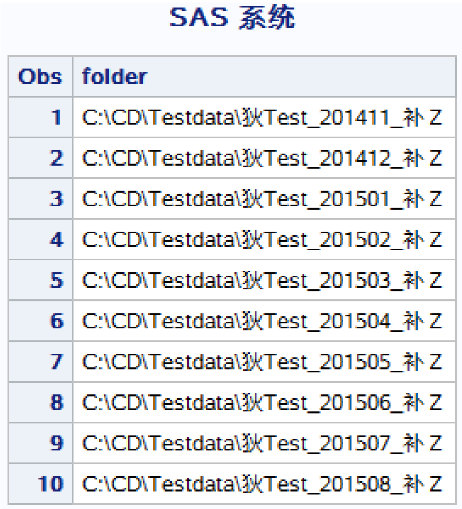
After processed by KCVT, the non-English characters encoded correctly.



VIEWTABLE: Work.A

	folder
1	C:\CD\Testdata\狄Test_201411_补Ｚ
2	C:\CD\Testdata\狄Test_201412_补Ｚ
3	C:\CD\Testdata\狄Test_201501_补Ｚ
4	C:\CD\Testdata\狄Test_201502_补Ｚ
5	C:\CD\Testdata\狄Test_201503_补Ｚ
6	C:\CD\Testdata\狄Test_201504_补Ｚ
7	C:\CD\Testdata\狄Test_201505_补Ｚ
8	C:\CD\Testdata\狄Test_201506_补Ｚ
9	C:\CD\Testdata\狄Test_201507_补Ｚ
10	C:\CD\Testdata\狄Test_201508_补Ｚ

Figure 3.7 Correct in dataset



SAS 系统

Obs	folder
1	C:\CD\Testdata\狄Test_201411_补Ｚ
2	C:\CD\Testdata\狄Test_201412_补Ｚ
3	C:\CD\Testdata\狄Test_201501_补Ｚ
4	C:\CD\Testdata\狄Test_201502_补Ｚ
5	C:\CD\Testdata\狄Test_201503_补Ｚ
6	C:\CD\Testdata\狄Test_201504_补Ｚ
7	C:\CD\Testdata\狄Test_201505_补Ｚ
8	C:\CD\Testdata\狄Test_201506_补Ｚ
9	C:\CD\Testdata\狄Test_201507_补Ｚ
10	C:\CD\Testdata\狄Test_201508_补Ｚ

Figure 3.8 Correct in print result

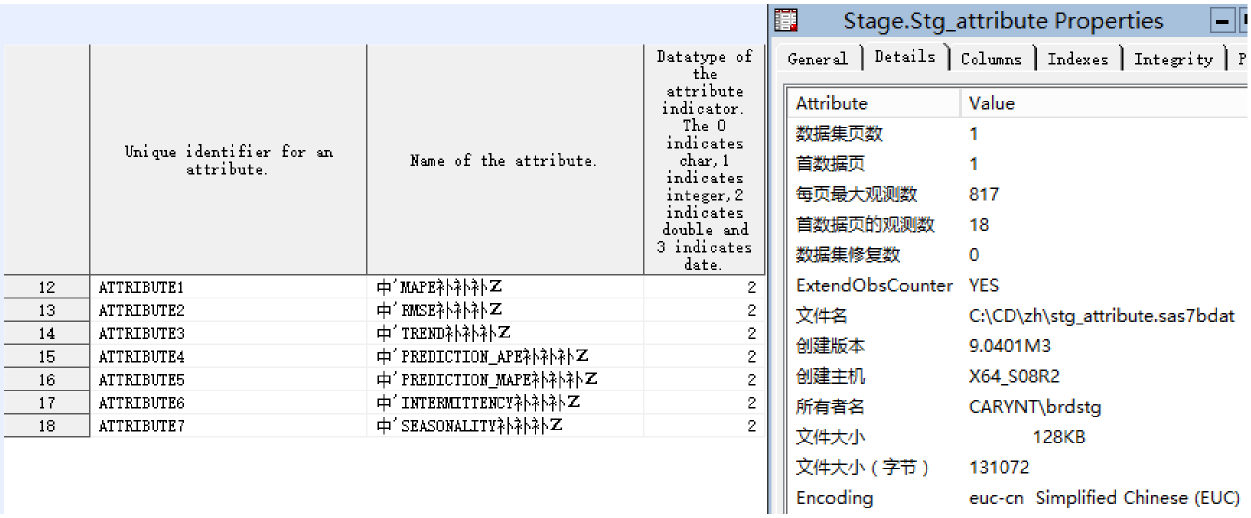
Similarly, we may encounter the same issue when input data from external files which are in vary encoding. For example, input data from a txt file with ANSI encoding in UTF-8 SAS session.

Conclusion: The encoding of data need our more attention.

TRANSCODING FOR SAS DATA SETS

Problem 3: Can the international datasets be read or modified correctly?

Besides the transcoding for the data got from OS, more of what we usually do the transcoding is for SAS data set. If the data set encoding does not match the SAS session encoding, processing data will meet error although non-utf-8 data set can be viewed in UTF-8 SAS session.

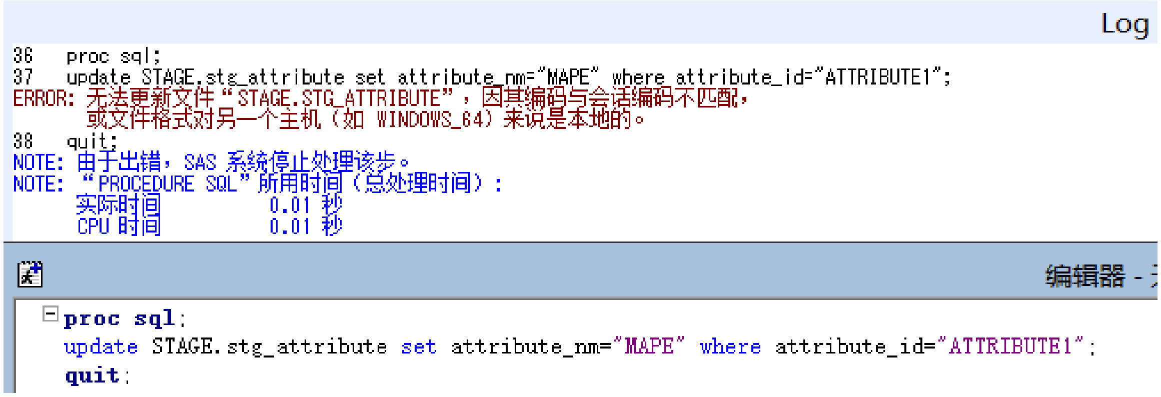


	Unique identifier for an attribute.	Name of the attribute.	Datatype of the attribute indicator. The 0 indicates char, 1 indicates integer, 2 indicates double and 3 indicates date.
12	ATTRIBUTE1	中'MAPE补补补Z	2
13	ATTRIBUTE2	中'RMSE补补补Z	2
14	ATTRIBUTE3	中'TREND补补补Z	2
15	ATTRIBUTE4	中'PREDICTION_APE补补补Z	2
16	ATTRIBUTE5	中'PREDICTION_MAPE补补补Z	2
17	ATTRIBUTE6	中'INTERMITTENCY补补补Z	2
18	ATTRIBUTE7	中'SEASONALITY补补补Z	2

Stage.Stg_attribute Properties

Attribute	Value
数据集页数	1
首数据页	1
每页最大观测数	817
首数据页的观测数	18
数据集修复数	0
ExtendObsCounter	YES
文件名	C:\CD\zh\stg_attribute.sas7bdat
创建版本	9.0401M3
创建主机	X64_S08R2
所有者名	CARYNT\brdstg
文件大小	128KB
文件大小（字节）	131072
Encoding	euc-cn Simplified Chinese (EUC)

Figure 4.1 View euc-cn data set in UTF-8 SAS session

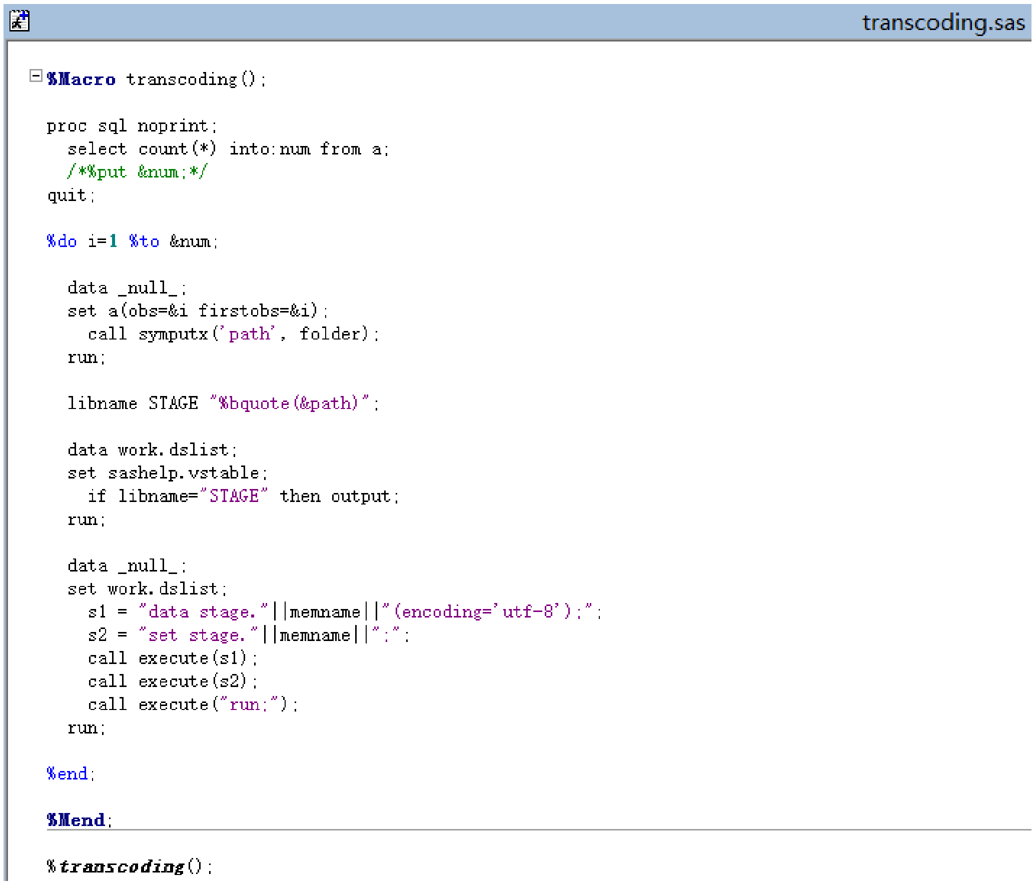


```
36    proc sql;
37    update STAGE.stg_attribute set attribute_nm="MAPE" where attribute_id="ATTRIBUTE1";
ERROR: 无法更新文件“STAGE.STG_ATTRIBUTE”，因其编码与会话编码不匹配，
       或文件格式对另一个主机（如 WINDOWS_64）来说是本地的。
38    quit;
NOTE: 由于出错，SAS 系统停止处理该步。
NOTE: “PROCEDURE SQL”所用时间（总处理时间）：
      实际时间          0.01 秒
      CPU 时间          0.01 秒
```

```
proc sql;
update STAGE.stg_attribute set attribute_nm="MAPE" where attribute_id="ATTRIBUTE1";
quit;
```

Figure 4.2 Process euc-cn data set in UTF-8 SAS session

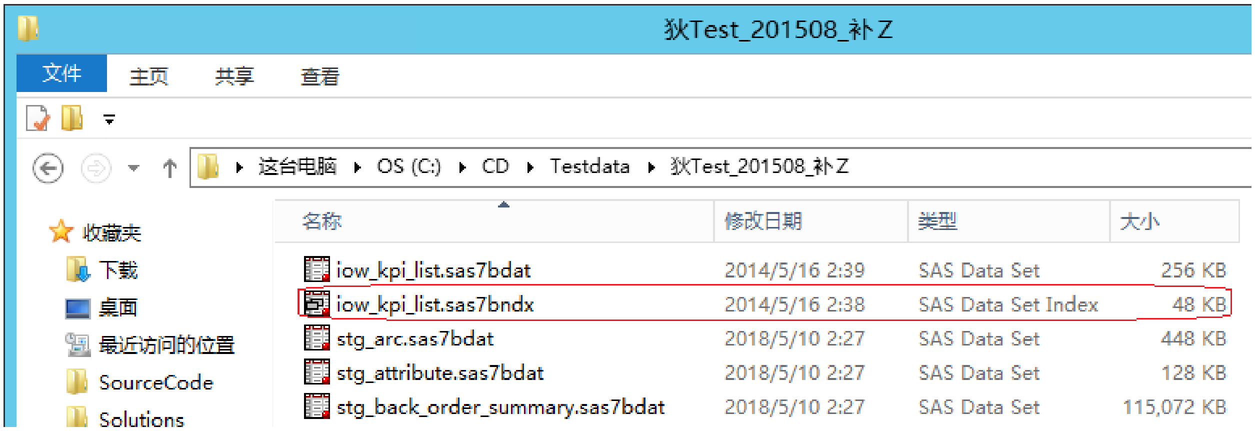
From SAS 9, data sets have an ENCODING attribute which is recorded in the file's descriptor information. Normally the ENCODING= data set option in data statement is used to override the encoding for a SAS data set.



```
%Macro transcoding();

proc sql noprint;
  select count(*) into:num from a;
  /*%put &num;*/
quit;

%do i=1 %to &num;

  data _null_;
  set a(obs=&i firstobs=&i);
    call symputx('path', folder);
  run;

  libname STAGE "%bquote(&path)";

  data work.dslist;
  set sashelp.vstable;
    if libname="STAGE" then output;
  run;

  data _null_;
  set work.dslist;
    s1 = "data stage."||memname||"(encoding='utf-8');";
    s2 = "set stage."||memname||";";
    call execute(s1);
    call execute(s2);
    call execute("run;");
  run;

%end;

%Mend;

%transcoding();
```

Figure 4.3 Use encoding= option in data statement

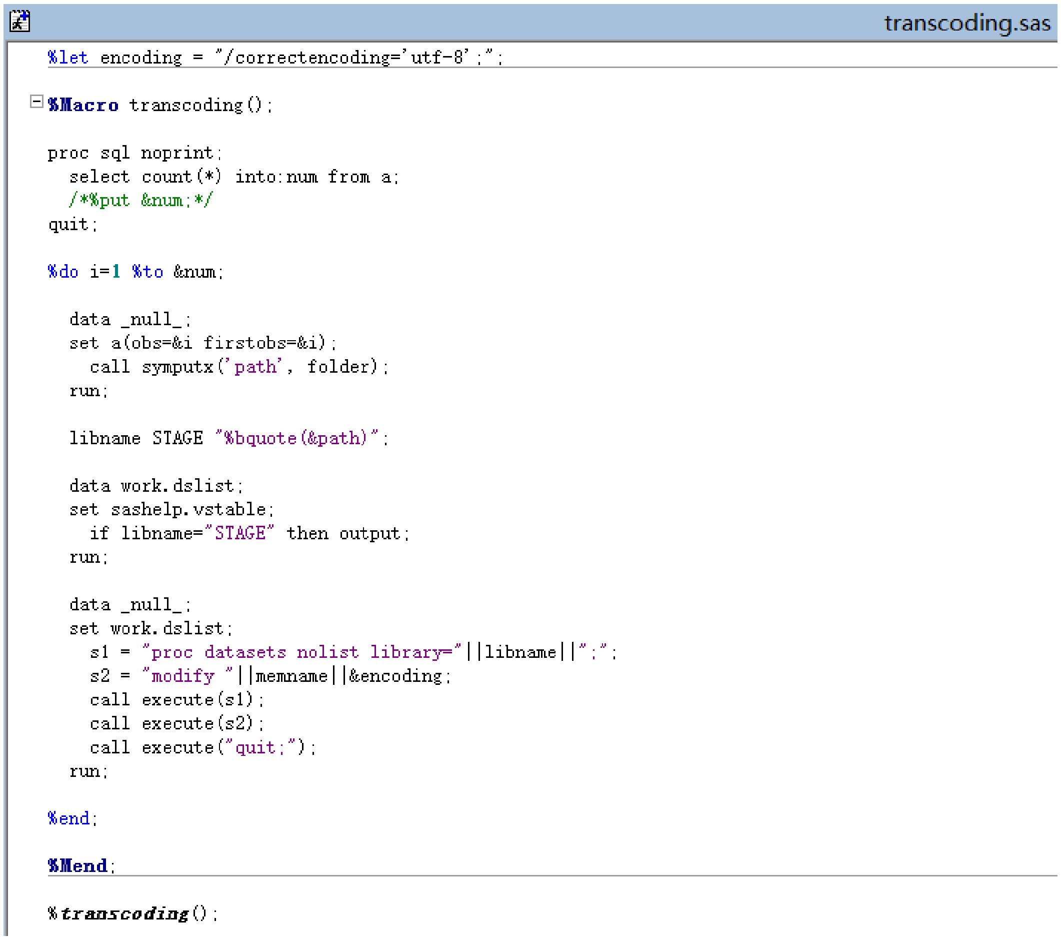
However, the transcoded data set which is generated by data statement with ENCODING= data set option will lose the index file (.sas7bndx file extension) if the source data set has indexes or integrity constraint.



名称	修改日期	类型	大小
iow_kpi_list.sas7bdat	2014/5/16 2:39	SAS Data Set	256 KB
iow_kpi_list.sas7bndx	2014/5/16 2:38	SAS Data Set Index	48 KB
stg_arc.sas7bdat	2018/5/10 2:27	SAS Data Set	448 KB
stg_attribute.sas7bdat	2018/5/10 2:27	SAS Data Set	128 KB
stg_back_order_summary.sas7bdat	2018/5/10 2:27	SAS Data Set	115,072 KB

Figure 4.4 INDEX file

So we recommend to use CORRECTENCODING= option in the MODIFY statement of the DATASETS procedure to do the transcoding for SAS data sets. CORRECTENCODING= option explicitly changes the encoding attribute of a SAS file to match the actual encoding of the data in the SAS file.



```
%let encoding = "/correctencoding='utf-8';";

%Macro transcoding();

proc sql noprint;
  select count(*) into:num from a;
  /*%put &num;*/
quit;

%do i=1 %to &num;

  data _null_;
  set a(obs=&i firstobs=&i);
    call symputx('path', folder);
  run;

  libname STAGE "%bquote(&path)";

  data work.dslist;
  set sashelp.vstable;
    if libname="STAGE" then output;
  run;

  data _null_;
  set work.dslist;
    s1 = "proc datasets nolist library="||libname||";";
    s2 = "modify "||memname||&encoding;
    call execute(s1);
    call execute(s2);
    call execute("quit;");
  run;

%end;

%Mend;

%transcoding();
```

Figure 4.5 CORRECTENCODING= option

CONCLUSION

If you met multilingual data process in SAS, The knowledge of encoding and transcoding is very important for processing international data. We must be clear how to solve this kind of issues, you can follow the guideline as below:

- Be clear the encoding of current SAS session
- Be clear the encoding of my data
- Do correctly and safely transcoding if actual data encoding does not match current SAS session encoding

REFERENCES

- **SAS Help and Documentation Viewer**. Copyright © 2018, SAS Institute Inc., Cary, NC, USA

CONTACT INFORMATION

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

Name: Roger (Di) Chen
Enterprise: SAS Beijing R&D
Address: 14/F Motorola Plaza, No.1 Wang Jing East Road, Chao Yang District, Beijing, China
City, State ZIP: Beijing, 100102
Work Phone: (8610) 8319 3355-3831
Fax: (8610) 8319 3355 / (8610) 6310 9130
E-mail: di.chen@sas.com

Web:
Twitter: