PHUSE US Connect 2021

14-18 June 2021
Paper DH12 – Data Handling (DH)

The Encoding Dilemma

Angelo Tinazzi, Cytel Inc.
Geneva – Switzerland
angelo.tinazzi@cytel.com

AGENDA

- Introduction
  - Encoding Dilemma?
- Encoding in a Nutshell
  - What is Encoding
  - Encoding Methods
- Encoding and SAS
  - Encoding vs Transcoding
  - Dealing with Transcoding in SAS
  - wlatin1 → UTF-8, UTF-8 → wlatin1, Failing transcode to wlatin1
  - Transcode datasets to your local encoding
  - Summary of Options
- Encoding and Data Submission
  - FDA, PMDA and NMPA Requirements
  - The SAS XPT Challenge with Multiple Byte Character Set (MBCS)
- Conclusions
  - Summary
  - Other aspects to consider → References

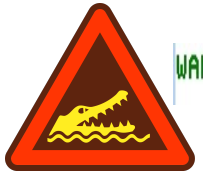
Introduction

Encoding Dilemma…..what’s that “weird” messages are telling me?

What's that "weird" messages are telling me?

```
NOTE: Data file SJISRAW.ADVEVE_1_0.DATA is in a format that is native to another host, or the file encoding does not match the session encoding.
      Cross Environment Data Access will be used, which might require additional CPU resources and might reduce
      performance.
```

Encoding?
Cross Environment Data Access?



```
WARNING: Some character data was lost during transcoding in the dataset SJISRAW.ADVEVE_1_0. Either the data contains characters that are not
         representable in the new encoding or truncation occurred during transcoding.
```

Transcoding?
Truncation?


```
ERROR: Some character data was lost during transcoding in the dataset TOCREATE.TESTTOUTF8.
Either the data contains characters
that are not representable in the new encoding or truncation occurred during transcoding.
NOTE: The DATA step has been abnormally terminated.
NOTE: The SAS System stopped processing this step because of errors.
NOTE: There were 2 observations read from the data set TOCREATE.TESTTOUTF8.
WARNING: The data set WORK.TESTTOUTF8 may be incomplete.  When this step was stopp
NOTE: DATA statement used (Total process time):
```

ERROR!!!

Encoding in a Nutshell

What is Encoding

Encoding Methods

What is Encoding?

Characters are stored as a series of bytes, where 1 byte = 8 bits

e.g. “Character” is intended a letter, a number, a symbol, etc.

“Sequence” of bits can represent an integer number

i.e. with 1 one byte, numbers between 0 and 255 (2^8-1=256)

Encoding is the way a computer interprets and represents the “Characters” within the data by assigning an integer number to each “Character” (code page)

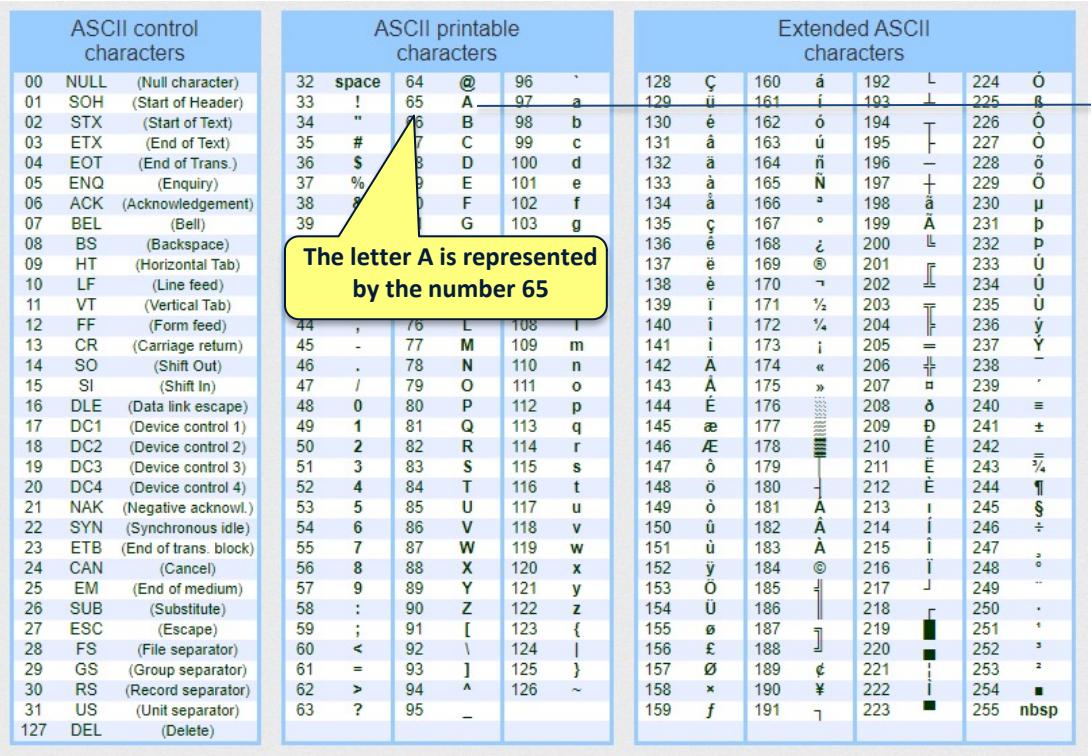
A **Character Set** is the set of characters used by a language or a group of language

e.g. English or more in general western languages, Chinese, Japanese, etc.

An **Encoding Method** is a set of rules that assign the numeric representations to the set of characters

SAS® 9.4 National Language Support (NLS): Reference Guide, Fifth Edition, SAS https://documentation.sas.com/api/collections/pgmsascdc/9.4_3.5/docsets/nlsref/content/nlsref.pdf?locale=en#nameddest=n04275wmuchnqjn1hfx0yzpopw5a

Encoding Methods Example WLATIN1 (Windows Latin 1, US and Western European)



ASCII control characters

00	NULL	(Null character)
01	SOH	(Start of Header)
02	STX	(Start of Text)
03	ETX	(End of Text)
04	EOT	(End of Trans.)
05	ENQ	(Enquiry)
06	ACK	(Acknowledgement)
07	BEL	(Bell)
08	BS	(Backspace)
09	HT	(Horizontal Tab)
10	LF	(Line feed)
11	VT	(Vertical Tab)
12	FF	(Form feed)
13	CR	(Carriage return)
14	SO	(Shift Out)
15	SI	(Shift In)
16	DLE	(Data link escape)
17	DC1	(Device control 1)
18	DC2	(Device control 2)
19	DC3	(Device control 3)
20	DC4	(Device control 4)
21	NAK	(Negative acknowl.)
22	SYN	(Synchronous idle)
23	ETB	(End of trans. block)
24	CAN	(Cancel)
25	EM	(End of medium)
26	SUB	(Substitute)
27	ESC	(Escape)
28	FS	(File separator)
29	GS	(Group separator)
30	RS	(Record separator)
31	US	(Unit separator)
127	DEL	(Delete)

ASCII printable characters

32	space	64	@	96	`
33	!	65	A	97	a
34	"	66	B	98	b
35	#	67	C	99	c
36	$	68	D	100	d
37	%	69	E	101	e
38	&	70	F	102	f
39	'	71	G	103	g
40	(	72	H	104	h
41	)	73	I	105	i
42	*	74	J	106	j
43	+	75	K	107	k
44	,	76	L	108	l
45	-	77	M	109	m
46	.	78	N	110	n
47	/	79	O	111	o
48	0	80	P	112	p
49	1	81	Q	113	q
50	2	82	R	114	r
51	3	83	S	115	s
52	4	84	T	116	t
53	5	85	U	117	u
54	6	86	V	118	v
55	7	87	W	119	w
56	8	88	X	120	x
57	9	89	Y	121	y
58	:	90	Z	122	z
59	;	91	[	123	{
60	<	92	\	124	|
61	=	93	]	125	}
62	>	94	^	126	~
63	?	95	_		

Extended ASCII characters

128	Ç	160	á	192	└	224	Ó
129	ü	161	í	193	┴	225	ß
130	é	162	ó	194	┬	226	Ô
131	â	163	ú	195	├	227	Ò
132	ä	164	ñ	196	─	228	õ
133	à	165	Ñ	197	┼	229	Õ
134	å	166	ª	198	ã	230	µ
135	ç	167	º	199	Ã	231	þ
136	ê	168	¿	200	╚	232	Þ
137	ë	169	®	201	╔	233	Ú
138	è	170	¬	202	╩	234	Û
139	ï	171	½	203	╦	235	Ù
140	î	172	¼	204	╠	236	ý
141	ì	173	¡	205	═	237	Ý
142	Ä	174	«	206	╬	238	¯
143	Å	175	»	207	¤	239	´
144	É	176	░	208	ð	240	≡
145	æ	177	▒	209	Ð	241	±
146	Æ	178	▓	210	Ê	242	‗
147	ô	179	│	211	Ë	243	¾
148	ö	180	┤	212	È	244	¶
149	ò	181	Á	213	ı	245	§
150	û	182	Â	214	Í	246	÷
151	ù	183	À	215	Î	247	¸
152	ÿ	184	©	216	Ï	248	°
153	Ö	185	╣	217	┘	249	¨
154	Ü	186	║	218	┌	250	·
155	ø	187	╗	219	█	251	¹
156	£	188	╝	220	▄	252	³
157	Ø	189	¢	221	¦	253	²
158	×	190	¥	222	Ì	254	■
159	ƒ	191	┐	223	▀	255	nbsp

The letter A is represented by the number 65

0	1	0	0	0	0	0	1
$0*2^7$	$1*2^6$	$0*2^5$	$0*2^4$	$0*2^3$	$0*2^2$	$0*2^1$	$1*2^0$

- With one byte (Single-Byte Character Set – SBCS) WLATIN1 can be represented
- This is sufficient for most of Western European Languages

https://theasciicode.com.ar/

Encoding Methods SBCS vs DBCS vs MBCS

A set of rules that assign the numeric representations to the set of characters

- Size of the encoding (type and number of characters represented)
- Number of bytes (bits) used
  - Single-Byte Character Set (**SBCS**)
  - Double-Byte Character Set (**DBCS**)
  - Multiple-Byte Character Set (**MBCS**)

Single-Byte Character Set (**SBCS**)

- **wlatin1** or **wlatin9** for Western European
- **wlatin2** for Eastern European

Double-Byte Character Set (**DBCS**)

- **Shift-Jis** for Japanese
- **Big5** or **EUC-CN** for Chinese

Encoding Methods SBCS vs DBCS vs MBCS

A set of rules that assign the numeric representations to the set of characters

- Size of the encoding (type and number of characters represented)
- Number of bytes (bits) used
  - Single-Byte Character Set (**SBCS**)
  - Double-Byte Character Set (**DBCS**)
  - Multiple-Byte Character Set (**MBCS**)

Multiple-Byte Character Set (**DBCS**)

- **UTF-8 Unicode** – Universal Character set Transformation
- An encoding that **attempt to represent all characters in all languages**
- **MBCS** (it uses 4 bytes)
- **>120000 characters.** For some 1 byte is still sufficient for many others not

Encoding and SAS

Encoding vs Transcoding

Dealing with Transcoding in SAS

wlatin1 → UTF-8, UTF-8 → wlatin1, Failing transcode to wlatin1

Transcode datasets to your local encoding

Summary of “Options”

Encoding vs Transcoding – Detecting your SAS Encoding

Encoding in SAS establishes the default working environment for your SAS session i.e. WLATIN1 in a Windows Os for US and Western European Languages

The encoding applied by default in your SAS environment **determines the default encoding of generated permanent datasets by your SAS programs i.e. SDTM**

Checking the encoding used by default by your SAS configuration

```
proc options option=encoding;
run;
```

```
 ENCODING=WLATIN1  Specifies the default character-set encoding for the SAS session.
NOTE: PROCEDURE OPTIONS used (Total process time):
      real time               0.01 seconds
      cpu time                0.01 seconds
```

Encoding vs Transcoding

Transcoding is the process of converting data from one encoding to another

The same character **could have different numeric representations in two different encodings**

SAS Cross-Environment Data Access (CEDA) in many cases does the transcoding for you

Checking the dataset encoding

```
/*Checking encoding*/
%macro encod(libin=,dsin=);
  %put;
  %put ----CHECKING FOR ENCODING----------------------------;
  %put       &libin..&dsin;
  %let dsid=%sysfunc(open(&libin..&dsin,i));
  %put       ENCODING is %sysfunc(attrc(&dsid,encoding));
  %put ------------------------------------------------------;
  %put;
  %put;
  %let dsclose=%sysfunc(close(&dsid));
%mend;
```

Checking the dataset encoding, 4 examples

- **UTF8** a SAS library with **SDTM** datasets generated in an **UTF-8 Unicode** encoding environment
- **SJID** a SAS library with **SDTM** datasets generated in a **shift-Jis Japanese** encoding environment
- **WLATIN** a SAS library with **SDTM** datasets generated in **WLatin1 western** encoding environment
- **SJISRAW** a SAS library with **legacy** datasets generated in **shift-Jis Japanese** encoding environment. **Some datasets contain some variables with Japanese characters**

My SAS session is using a WLatin1 western encoding

Checking the dataset encoding

```
232  %encod(libin=utf8,dsin=ae);

----CHECKING FOR ENCODING----------------------------
utf8.ae
NOTE: Data file UTF8.AE.DATA is in a format that is native to another host, or the file encoding does not match the session encoding.
      performance.
ENCODING is utf-8  Unicode (UTF-8)
---------------------------------------------------
```

Transcoding Successful

```
233  %encod(libin=SJIS,dsin=ae);

----CHECKING FOR ENCODING----------------------------
SJIS.ae
NOTE: Data file SJIS.AE.DATA is in a format that is native to another host, or the file encoding does not match the session encoding.
      performance.
ENCODING is shift-jis  Japanese (SJIS)
---------------------------------------------------
```

Transcoding Successful

```
234  %encod(libin=wlatin,dsin=Adveve_1_0);

----CHECKING FOR ENCODING----------------------------
wlatin.Adveve_1_0
ENCODING is wlatin1  Western (Windows)
---------------------------------------------------
```

Same encoding, no need for transcoding

Dealing with Transcoding in SAS

Opening a dataset with **shift-Jis Japanese** (**SJIS**) encoding with Japanese Characters in a **WLatin1** SAS encoding environment

VIEWTABLE: Sjisraw.Adveve_1_0 (Pretreatment/Adverse Events)

	AESEQ	AEVTJ	AESDT	AEEDT
1	1	→→→	11APR2012	30MAY2012
2	1	→→→→→	21MAR2012	26MAR2012
3	1	→→	21MAR2012	22MAR2012
4	1	→→→→→→→→→→	21FEB2012	06MAR2012
5	2	→→→→→	26FEB2012	29NOV2012
6	2	→→→→→	17SEP2012	08OCT2012
7	1	AST→→	21SEP2012	19NOV2012

CEDA in action

```
NOTE: Data file SJISRAW.ADVEVE_1_0.DATA is in a format that is native to another host, or the file encoding does not match the session encoding.
      Cross Environment Data Access will be used, which might require additional CPU resources and might reduce
      performance.

WARNING: Some character data was lost during transcoding in the dataset SJISRAW.ADVEVE_1_0. Either the data contains characters that are not
         representable in the new encoding or truncation occurred during transcoding.
```

Transcoding Failed

Dealing with Transcoding in SAS

When **transcoding fails**, the only solution is to have your **SAS session using the same Encoding of the original SAS session (or use UTF-8)**, this is determined at **SAS startup (AUTOEXEC)**

```
-ENCODING UTF-8
```

Before with WLATIN1

VIEWTABLE: Sjisraw.Adveve_1_0 (Pretreatment/Adverse Events)

	AESEQ	AEVTJ
1	1	→→→
2	1	→→→→→
3	1	→→
4	1	→→→→→→→→→→
5	2	→→→→→
6	2	→→→→→
7	1	AST→→

Now with UTF-8

VIEWTABLE: Sjisraw.Adveve_1_0 (Pretreatment/Adverse Events)

	AESEQ	AEVTJ	AESDT	AEEDT
1	1	脳梗塞	11APR2012	30MAY2012
2	1	急性膀胱炎	21MAR2012	26MAR2012
3	1	下痢	21MAR2012	22MAR2012
4	1	マロリーワイス症候群	21FEB2012	06MAR2012
5	2	円形脱毛症	26FEB2012	29NOV2012
6	2	便秘の悪化	17SEP2012	08OCT2012
7	1	AST上昇	21SEP2012	19NOV2012

LButf8 a dataset generated with UTF-8 encoding

VIEWTABLE: Transf1.Lbwlatin1

	unit
1	µmol/l
2	mmol/l
3	g/l

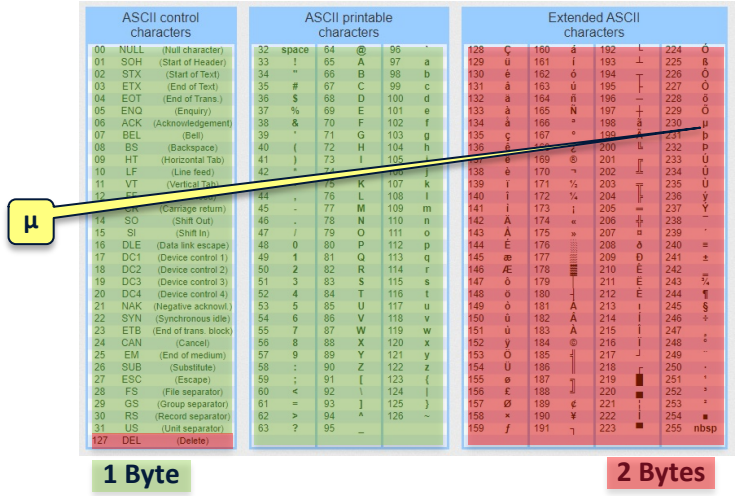
Transf1.Lbwlatin1 Properties

General | Details | Columns | Indexes | Integrity

Find column name:

Column Name	Type	Length	Fo
Aa unit	Text	6	

UTF-8 Extended ASCII Representation



Reading LButf8 in an wlatin1 encoding SAS session

```
data libutf8.LBUTF8;
  set libwlat.LBwlatin1;
run;
```

VIEWTABLE: Libutf8.Lbutf8

	unit	unit_len
1	?mol/l	6
2	mmol/l	6
3	g/l	

The "µ" is not correctly transcoded

µ

1	0	1	1	0	1	0	1

wlatin1 (1 Byte)
SAS length=1

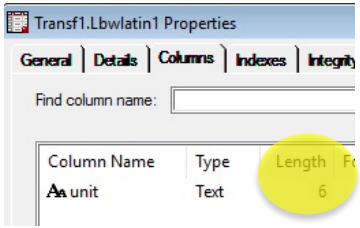
1	1	0	0	0	0	1	0	1	0	1	1	0	1	0	1

utf-8 (2 Bytes)
SAS length=2

LBwlatin1 a dataset generated with WLATIN1 encoding

VIEWTABLE: Transf1.Lbwlatin1

	unit
1	µmol/l
2	mmol/l
3	g/l



Reading LBwlatin1 in an UTF-8 encoding SAS session

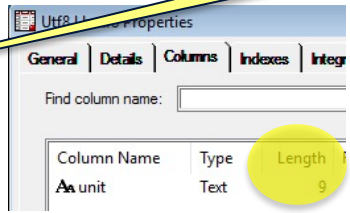
Try the CVP libname option

```
libname libwlat  cvp "C:\Users\Angelo.Tinaz
data utf8.LBUTF8;
  set libwlat.LBwlatin1;
run;
```

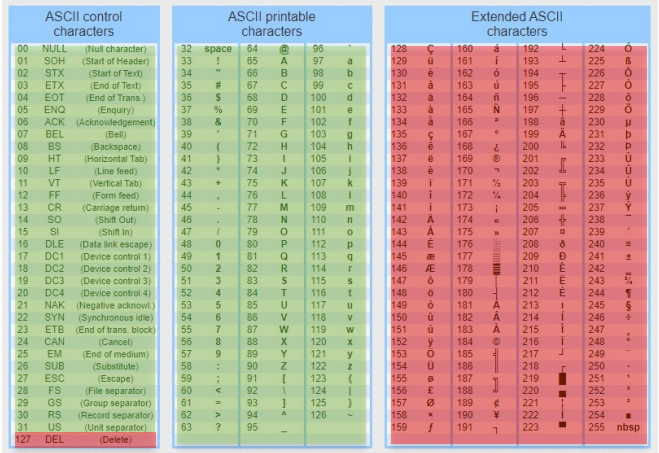
The "µ" is now correctly transcoded

VIEWTABLE: Utf8.Lbutf8

	unit
1	µmol/l
2	mmol/l
3	g/l



UTF-8 Extended ASCII Representation



LButf8 a dataset generated with UTF-8 encoding

VIEWTABLE: Libutf8.Lbutf8

	unit
1	µmol/l
2	mmol/l
3	g/l

Reading LButf8 in an wlatin1 encoding SAS session

VIEWTABLE: Work.Lbwlatin1

	unit
1	µmol/l
2	mmol/l
3	g/l

```
NOTE: Data file LIBUTF8.LBUTF8.DATA is in a format
that …… Cross Environment Data Access will be used……
```

The transcoding was successful, no errors

Failing transcode to wlatin1

SJISRaw a dataset generated with SJIS encoding containing Japanese characters

```
data Adveve_1_0;
  set SJISRaw.Adveve_1_0;
run;
ERROR: Some character data was lost during transcoding in the dataset SJISRAW.ADVEVE_1_0. Either the
NOTE: The DATA step has been abnormally terminated.
NOTE: The SAS System stopped processing this step because of errors.
WARNING: The data set WORK.ADVEVE_1_0 may be incomplete.  When this step was stopped there were 0 obs
WARNING: Data set WORK.ADVEVE_1_0 was not replaced because this step was stopped.
```

```
data Adveve_1_0;
  set SJISRaw.Adveve_1_0(encoding=ANY);
run;
NOTE: There were 212 observations read from the data set SJISRAW.ADVEVE_1_0.
NOTE: The data set WORK.ADVEVE_1_0 has 212 observations and 50 variables.
```

Failing transcode to wlatin1

LButf8 a dataset generated with UTF-8 encoding

VIEWTABLE: Libutf8.Lbutf8

	unit
1	µmol/l
2	mmol/l
3	g/l

Reading LButf8 in an wlatin1 encoding SAS session with encoding=ANY option

VIEWTABLE: Work.Lbwlatin1

	unit
1	Âµmol/l
2	mmol/l
3	g/l

```
data lbwlatin1;
  set libutf8.lbutf8(encoding=any);
run;
```

Failing transcode to wlatin1

LButf8 a dataset generated with UTF-8 encoding

VIEWTABLE: Libutf8.Lbutf8

	unit
1	µmol/l
2	mmol/l
3	g/l

Reading LButf8 in an wlatin1 encoding SAS session with encoding=ANY option

VIEWTABLE: Work.Lbwlatin1

	unit
1	µmol/l
2	mmol/l
3	g/l

```
data lbwlatin1;
  set libutf8.lbutf8(encoding=ANY);
  array allchar (*) _character_;
  do i=1 to dim(allchar);
     txt=KCVT(allchar(i),"UTF-8","WLATIN1");
     varname=vname(allchar(i));
     if txt^=allchar(i) then do;
        put ".... Instring for variable " varname " : " allchar(i) "converted to: " txt;
        allchar(i)=txt;
     end;
  end;
  drop txt varname;
run;
```

```
.... Instring for variable unit  : Âµmol/l converted to: µmol/l
```

Transcode datasets to your local encoding

SAS CEDA transcode for you

```
NOTE: Cross Environment Data Access will be used,
which might require additional CPU resource and might
reduce performance
```

But it requires additional CPU resources, Time!!!!

```
proc copy inlib=sjis outlib=transf2 noclone;
run;
```

```
/*Test use of CPU: Using original datasets */
proc sql noprint;
  create table testSQLOriginalEncoding as
    select a.*, b.arm, b.race, b.sex
    from SJIS.AE a left join SJIS.DM b
    on a.usubjid=b.usubjid;
quit;
```

```
NOTE: PROCEDURE SQL used (Total process time):
      real time               0.13 seconds
      cpu time                0.06 seconds
```

```
/*Test use of CPU: Using transcoded datasets */
proc sql noprint;
  create table testSQLTanscodedEncoding as
    select a.*, b.arm, b.race, b.sex
    from TRANSF2.AE a left join TRANSF2.DM b
    on a.usubjid=b.usubjid;
quit;
```

```
NOTE: PROCEDURE SQL used (Total process time):
      real time               0.06 seconds
      cpu time                0.04 seconds
```

Encoding and Data Submission

FDA, PMDA and NMPA Requirements

The SAS XPT Challenge with Multiple Byte Character Set (MBCS)

FDA

From the FDA Study Data Technical Conformance Guide section 3.3.5

Variable names, as well as variable and dataset labels should include ***American Standard Code for Information Interchange (ASCII) text codes only****. Variable values are the most broadly compatible with software and operating systems when they are* ***restricted to ASCII text codes (printable values below 128)****. Use UTF-8 for extending character sets; however,* ***the use of extended mappings is not recommended.*** ***Transcoding errors, variable length errors,*** *and* ***lack of software support for multi byte UTF-8 encodings*** *can result in* ***incorrect character display*** *and variable value* ***truncations****.* ***Ensure that LBSTRESC*** *and controlled terminology extensions in* ***LBTEST*** *do not contain byte values* ***160-191*** *as some character mappings in that range may interfere with agency processes.*

FDA

restricted to ASCII text codes (printable values below 128)

ASCII control characters		
00	NULL	(Null character)
01	SOH	(Start of Header)
02	STX	(Start of Text)
03	ETX	(End of Text)
04	EOT	(End of Trans.)
05	ENQ	(Enquiry)
06	ACK	(Acknowledgement)
07	BEL	(Bell)
08	BS	(Backspace)
09	HT	(Horizontal Tab)
10	LF	(Line feed)
11	VT	(Vertical Tab)
12	FF	(Form feed)
13	CR	(Carriage return)
14	SO	(Shift Out)
15	SI	(Shift In)
16	DLE	(Data link escape)
17	DC1	(Device control 1)
18	DC2	(Device control 2)
19	DC3	(Device control 3)
20	DC4	(Device control 4)
21	NAK	(Negative acknowl.)
22	SYN	(Synchronous idle)
23	ETB	(End of trans. block)
24	CAN	(Cancel)
25	EM	(End of medium)
26	SUB	(Substitute)
27	ESC	(Escape)
28	FS	(File separator)
29	GS	(Group separator)
30	RS	(Record separator)
31	US	(Unit separator)
127	DEL	(Delete)

ASCII printable characters					
32	space	64	@	96	`
33	!	65	A	97	a
34	"	66	B	98	b
35	#	67	C	99	c
36	$	68	D	100	d
37	%	69	E	101	e
38	&	70	F	102	f
39	'	71	G	103	g
40	(	72	H	104	h
41	)	73	I	105	i
42	*	74	J	106	j
43	+	75	K	107	k
44	,	76	L	108	l
45	-	77	M	109	m
46	.	78	N	110	n
47	/	79	O	111	o
48	0	80	P	112	p
49	1	81	Q	113	q
50	2	82	R	114	r
51	3	83	S	115	s
52	4	84	T	116	t
53	5	85	U	117	u
54	6	86	V	118	v
55	7	87	W	119	w
56	8	88	X	120	x
57	9	89	Y	121	y
58	:	90	Z	122	z
59	;	91	[	123	{
60	<	92	\	124	|
61	=	93	]	125	}
62	>	94	^	126	~
63	?	95	_		

Removing “control characters”

```
mychar=compress(mychar,,'kw');
```

Other “control characters”

FDA

LBTEST** do not contain byte values **160-191

Extended ASCII characters

128	Ç	160	á	192	└	224	Ó
129	ü	161	í	193	┴	225	ß
130	é	162	ó	194	┬	226	Ô
131	â	163	ú	195	├	227	Ò
132	ä	164	ñ	196	─	228	õ
133	à	165	Ñ	197	┼	229	Õ
134	å	166	ª	198	ã	230	µ
135	ç	167	º	199	Ã	231	þ
136	ê	168	¿	200	╚	232	Þ
137	ë	169	®	201	╔	233	Ú
138	è	170	¬	202	╩	234	Û
139	ï	171	½	203	╦	235	Ù
140	î	172	¼	204	╠	236	ý
141	ì	173	¡	205	═	237	Ý
142	Ä	174	«	206	╬	238	¯
143	Å	175	»	207	¤	239	´
144	É	176	▒	208	ð	240	≡
145	æ	177	▒	209	Ð	241	±
146	Æ	178	▒	210	Ê	242	‗
147	ô	179	│	211	Ë	243	¾
148	ö	180	┤	212	È	244	¶
149	ò	181	Á	213	ı	245	§
150	û	182	Â	214	Í	246	÷
151	ù	183	À	215	Î	247	¸
152	ÿ	184	©	216	Ï	248	°
153	Ö	185	╣	217	┘	249	¨
154	Ü	186	║	218	┌	250	·
155	ø	187	╗	219	▉	251	¹
156	£	188	╝	220	▄	252	³
157	Ø	189	¢	221	¦	253	²
158	×	190	¥	222	Ì	254	■
159	ƒ	191	┐	223	▀	255	nbsp

Use of Extended ASCII characters require careful investigation → no automatic replacement

Non-Printable and Special Characters? … BYTE me!

L. Sims, PHUSE 2016

PMDA (Japan) and NMPA (China)

- **Some variables need to be provided in both English and Japanese/Chinese i.e. AE verbatim**
- NMPA have additional requirements
  - **Define-xml and Reviewer Guide in Chinese**
  - **Encoding used to be specified in the reviewer guide**
- **SAS Datasets and Variables Labels in Chinese**
- SAS XPT is still the data transfer format (not necessarily XPT 5 for NMPA)

The SAS XPT Challenge with Multiple Byte Character Set (MBCS)

- Some variables need to be provided in both English and Japanese/Chinese i.e. AE verbatim
- NMPA have additional requirements
  - Define-xml and Reviewer Guide in Chinese
  - **SAS Datasets Labels in Chinese**
  - **SAS XPT is SBCS**

Some SDTM Labels won't fit!

CMTRT: *Reported Name of Drug, Med, or Therapy*
Chinese: 报告药品报告药品名称,药物治疗
requires 42 bytes, available are only 40 - last character will be lost

HODECOD: *Dictionary-Derived Term for the Healthcare Encounter*
Chinese: 词典中针对医疗保健遇到的术语
requires 42 bytes, available are only 40 - last character will be lost

http://xml4pharma.com/publications/Poster_Jozef_Aerts_Chinese_characters_XPT.pdf

Conclusions

Summary

Other aspects to consider → References

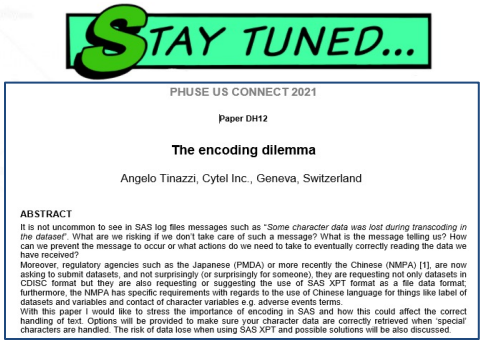
Summary

Make sure the encoding of the data you receive is compatible with your encoding and apply the needed **transformations**

- CDISC-CT, US-FDA requirements
- Other Agencies requirements

Other Aspects to consider

- Length of variables when encoding is not SBCS
- Working with MBCS Encoding (Use of “K” functions)



Other aspects to consider → References

- **Tips and Fixes for Cross-Environment Batch Transfer of SAS® Data** – Y. Zhuo; PharmaSUG 2018
- **Data Encoding: All Characters for All Countries** – D. Dutton; PHUSE 2015
- **The impact of Change from wlatin1 to UTF-8 in SAS Environment** – H. Song, A. Koster; PharmaSUG 2016
- **UTF What? A Guide for Handling SAS Transcoding Errors with UTF-8 Encoded Data** – M. Stackhouse, L. Pogula, PharmaSUG 2018
- **SAS 9.3 UTF-8 Encoding Support and Related Issue Troubleshooting** – J. Liang, Edmonton User Group 2016
- **Non-Printable and Special Characters? … BYTE me!** – L. Sims, PHUSE 2016
- **SAS® 9.4 National Language Support (NLS): Reference Guide**, Fifth Edition, SAS
  https://documentation.sas.com/api/collections/pgmsascdc/9.4_3.5/docsets/nlsref/content/nlsref.pdf?locale=en#nameddest=n04275wmuchnqjn1hfx0yzpopw5a
  - Chapter 3 “Encoding for NLS”
  - Chapter 4 “Transcoding for NLS”
  - Chapter 5 “Double-Byte Characters Sets (DBCS)”
- **Guideline on the Submission of Clinical Trial**, NMPA Center for Drug Evaluation Data (https://www.nmpa.gov.cn/directory/web/nmpa/images/obbSqc7vwdm0ssrU0enKb7dtd29u9a4tbzUrdTyo6jK1NDQo6mhty5wZGY=.pdf)
- **Chinese and Asian characters in SAS Transport 5 datasets: Why that is the worst possible choice**, J. Aerts, 2020 (unpublished CDISC-US Interchange poster) http://xml4pharma.com/publications/Poster_Jozef_Aerts_Chinese_characters_XPT.pdf

Angelo Tinazzi, Senior Director
Cytel Inc.
Standards, Systems, CDISC Consulting, Statistical Programming
Clinical Research Services
Route de Pré-Bois 20
C.P 1839, 1215 Geneva, SWITZERLAND
Email: angelo.tinazzi@cytel.com

Backup Slides

Getting more details with PROC CONTENTS (dataset with UTF-8 encoding)

```
proc contents data=utf8.ae;
run;
```

Encoding Details for utf8.ae

The CONTENTS Procedure

Data Set Name	UTF8.AE	Observations	200
Member Type	DATA	Variables	31
Engine	V9	Indexes	0
Created	05/22/2018 23:39:22	Observation Length	648
Last Modified	05/22/2018 23:39:22	Deleted Observations	0
Protection		Compressed	NO
Data Set Type		Sorted	NO
Label	Adverse Events		
Data Representation	HP_UX_64, RS_6000_AIX_64, SOLARIS_64, HP_IA64		
Encoding	utf-8 Unicode (UTF-8)		

Different encoding
Different Host/Os

Getting more details with PROC CONTENTS (dataset with SJIS encoding)

```
proc contents data=SJIS.ae;
run;
```

Encoding Details for SJIS.ae

The CONTENTS Procedure

Data Set Name	SJIS.AE	Observations	1137
Member Type	DATA	Variables	33
Engine	V9	Indexes	0
Created	10/24/2019 04:10:06	Observation Length	672
Last Modified	10/24/2019 04:10:06	Deleted Observations	0
Protection		Compressed	NO
Data Set Type		Sorted	NO
Label	Adverse Events		
Data Representation	WINDOWS_64		
Encoding	shift-jis Japanese (SJIS)		

Different encoding
Same Host/Os

Getting more details with PROC CONTENTS (dataset with wlatin1 encoding)

```
proc contents data=wlatin.Adveve_1_0;
run;
```

Encoding Details for wlatin.Adveve_1_0

The CONTENTS Procedure

Data Set Name	WLATIN.ADVEVE_1_0	Observations	314
Member Type	DATA	Variables	50
Engine	V9	Indexes	0
Created	05/20/2020 06:42:11	Observation Length	4568
Last Modified	05/20/2020 06:42:11	Deleted Observations	0
Protection		Compressed	NO
Data Set Type		Sorted	NO
Label			
Data Representation	WINDOWS_64		
Encoding	wlatin1 Western (Windows)		

Same encoding
Same Host/Os

Encoding Methods SBCS vs DBCS vs MBCS

A set of rules that assign the numeric representations to the set of characters

- Size of the encoding (type and number of characters represented)
- Number of bytes (bits) used
  - Single-Byte Character Set (**SBCS**)
  - Double-Byte Character Set (**DBCS**)
  - Multiple-Byte Character Set (**MBCS**)

Encoding	Character Set	Size (Nr. of Bytes)
WLATIN1 (SBC)	ASCII (and extended ASCII)	1
SHIFT-JIS	ASCII, Katakana, other Japanese	1 or 2 bytes
UTF-8	ASCII, foreign languages, special symbols and more	1 to 4 bytes

Summary of "Options"

Checking your SAS session encoding

```
proc options option=encoding;
run;
```

Checking dataset encoding

```
%let dsid=%sysfunc(open(<libname>.<dataset name>,i));
%let encodDS=%sysfunc(%attrc(&dsid,encoding));
```

Changing your SAS session encoding (AUTOEXEC)

```
-encoding UTF-8
```

If you need to switch to a different session

Avoiding data truncation from wlatin1 to UTF-8

```
libname libor cvp "<original datasets folder>";
proc copy inlib=libor outlib=libnew noclone;
run;
```

Avoiding errors when reading UTF-8 datasets in wlatin1 session

```
set <UTF-8 dataset>(encoding=ANY);
libname libor "<original datasets folder>"
        inencoding=ANY;
```

Assessing and trying to correct encoding=ANY

```
<var>=KCVT(<var>,"<original encoding>",
                 "<current session encoding>");
i.e. myvar=KCVT(myvar,"SJIT","WLATIN1");
```