Failed to transcode data from U_UTF8_CE to U_LATIN9_CE ?!

15 January, 2014 - 16:51 — Student-Swen

Liebe redscope-Freunde,

ich habe mal wieder eine Frage an euch zu der ich über die Suchfunktion und google leider passende Hilfe gefunden habe. Ich hoffe ihr könnt mir helfen!

Ich arbeite mit Daten auf die ich nur per "kontrollierter Datenfernverarbeitung" zugreifen kann. In der Praxis sieht es so aus, dass ich im Büro einen Berg Code schreibe und diese auf einem synthetischen Datensatz auf meinem Windows-Rechner teste. Läuft der Code durch, schicke ich diese zum "Dateninhaber". Ein Mitarbeiter vor Ort führt dann meine Codes auf dem Original-Datensatz aus. Bei denen läuft SAS aber auf einem UNIX-Server, was anscheind teilweise zu Problemen führt.

Beim Dateninhaber gibt SAS die Fehlermeldung aus dem Anhang aus.

Kann mir einer erklären was das für ein Fehler ist und wie ich meinen Code so ändere, dass er beim Dateninhaber auf dem UNIX-System läuft?

Vielen Dank für eure Hilfe
Swen

Anhang	Größe
SASerror.jpg	65.34 KB

Foren:
Allgemeine Fragen zu SAS

Sprach- bzw. Zeichensatz-Einstellung

16 January, 2014 - 11:18 — HansKneilmann

Hallo Swen,
meiner Meinung nach liegt das nicht daran, dass der Windows-Rechner-Code auf einem UNIX-Server laufen soll, sondern an dem Sprach-bzw. Zeichensatz-Einstellungen beider Rechner.
Die lange Fehlermeldung sagt aus, dass SAS ein Problem beim übersetzen von UTF8 nach Latin9 hat.
UTF8 kenne ich auch als Unicode und bei Unicode ist so ziemlich alles an Sonderzeichen möglich, was man sich so vorstellen kann. Im Gegensatz dazu ist Latin9 nur ein etwas aufgepeppter ASCII-Zeichensatz.
Man nennt Latin9 auch Westeuropäisch, weil dort z.B. die französischen Spezial-Buchstaben enthalten sind.
Es fehlen z.B. die polnischen oder die ungarischen Spezial-Buchstaben, die sind in Latin2 enthalten, aber dort fehlen die französischen ....
Im UTF8-Zeichensatz sind sie alle drin, auch die Luxus-Bindestriche (EM-Dash und EN-Dash) oder die Luxus-Gänsefüßchen („Text‟ statt "Text" und ‚Text‘ statt 'Text') die man z.B. mit Word in Texte

einfügen kann.
Beim Umsetzen von Texten bzw. von Programm-Code, der in UTF8 codiert ist in eine Datei, die in Latin9 codiert ist gibt es logischerweise Zeichen, die nicht umsetzbar sind. Genau diese produzieren die oben genannte Fehlermeldung.
Die Abhilfe ist eigentlich ganz einfach: Bei Systeme müssen die gleiche Codierung verwenden. Entweder beide UTF8 (bzw. U_UTF8_CE) oder beide Latin9 (bzw. U_LATIN9_CE).
Oder im Quellsystem muss man peinlich genau darauf achten keine falschen Zeichen zu verwenden.
Oder schon beim exportieren der Daten bzw. des Codes aus dem Quellsystem die Umcodierung machen, dadurch tritt der Transcoding-Fehler früher, beim Verursacher, auf und kann im Quellsystem behoben werden.
Beispiel Einlesen einer Latin2-codierten-Datei:

```
infile Datei LRECL=777 dlm=";" pad missover encoding="latin2" firstobs=
```

Beispiel Schreiben einer Latin2-codierten-Datei:

```
file  "&filename." encoding="latin1";
```

Beispiel einer Unicode- bzw. UTF8-Libref (alle Data Sets dort sind autom. UTF8-codiert):

```
libname dmadm_u8 "&XXDIR._u8" outencoding="utf-8";
```

Beispiel für die Ziel-Zeile im Start-Ikon einer 'normalen' SAS-Session auf einem Win7-PC:
C:\DWH\SAS93\x86\SASFoundation\9.3\sas.exe -CONFIG
C:\DWH\SAS93\x86\SASFoundation\9.3\nls\en\SASV9.CFG
Beispiel für den Start einer 'Unicode' SAS-Session auf einem Win7-PC
C:\DWH\SAS93\x86\SASFoundation\9.3\sas.exe -CONFIG
C:\DWH\SAS93\x86\SASFoundation\9.3\nls\u8\SASV9.CFG
(man beachte den kleinen feinen Unterschied: nls\u8 statt nls\en)

Gruß
Hans Kneilmann, Schäfer Shop GmbH (SSI)

P.S.: Unser Daten lieferndes System ist unicode-fähig und unser (SAS-) DWH ist es nicht, d.h. mit der Umcodierung bzw. mit dem Transcoding-Fehler habe ich länger gekämpft, bis alles zufriedenstellend lief.

Fehler bereits beim Laden einiger Makros

16 January, 2014 - 09:39 — Student-Swen

Hallo Herr Kneilmann,

vielen Dank für die prompte Hilfe!

Leider verstehe ich ihre Codes nur bedingt. Ich verstehe aber das Problem.

Eine Sache ist bei meinem ersten Post nicht so klar geworden wie ich dachte:
Der Fehler wird nicht erst beim schreiben eines Datensatzes angezeigt, sondern bereits beim einlesen einiger Makros. Ich habe mein Projekt in einzelne Module unterteilt und "missbrauche" dafür Makros. Vereinfacht sieht der Code wie folgt aus.

```
/*Schritt 1: Makros einlesen*/
%macro A;
code...
%mend;

...

%macro Z;
code...
%mend;

/*Schritt 2: Makros ausführen*/
data bib.data_neu;
set bib.data_alt;

%A;
...
%Z;

run;
```

Das ganze halt nur auf knapp 35.000 Codezeilen, was eine händische Kontrolle unmöglich macht.
Der Fehler wird bereits beim ersten Schritt (Makros einlesen) angezeigt. D.h. im Code selbst müssen schon Sonderzeichen enthalten sein, die bei der ausführenden Stelle nicht verarbeitet werden können.

Der Code ist zu 99% selbst geschrieben. Sonderzeichen verwende ich in SAS generell nicht. Allerdings sind die Variablenlabels aus einer Excel-Datei (Datensatzbeschreibung) per Copy & Paste übernommen worden. Ich vermute, dass der Fehler daher kommt.

Gibt es eine Möglichkeit nach diesen Sonderzeichen gezielt zu suchen und diese zu löschen? Ich kann leider keinen Einfluss darauf nehmen, wie SAS bei der ausführenden Stelle gestartet wird oder welcher Zeichensatz verwendet wird.

Viele Grüße
Swen

Lösungs-Ansätze

16 January, 2014 - 10:41 — HansKneilmann

Hallo Swen,
das es im vorliegenden Fall um den Programm-Code und nicht um Daten geht ist schon klar geworden, aber *meine* Erfahrung bezieht sich nur auf Daten, nicht auf Code.
Aber Programm-Code ist auch *nur* Text und Texte sind natürlich auch nichts anderes als Daten ...

Das Problem mit der Menge an Codezeilen kenne ich gut: Hier bei uns gibt es so ca. 595.000 Codezeilen und das größte Einzelstück hat ca. 50.000 Zeilen (eigentlich mehr, einiges ist ausgelagert ...) und ist auch in Macros unterteilt (kein Missbrauch, sondern eine sinnvolle Sache!).

Auch klar war, dass das Ziel-System außerhalb Deines Einflussbereiches liegt und Du die Lösung dort nicht findest.

Der Fehler wird bereits beim ersten Schritt (Makros einlesen) angezeigt. D.h. im Code selbst

müssen schon Sonderzeichen enthalten sein, (..) Auch das war schon aus dem 1. Beitrag klar.

Zurück zu den Lösungs-Ansätzen:
Du arbeitest mit dem SAS-EG. Ich würde den EG nicht mehr im Unicode-Modus sondern 'normal' starten. Beim *alten* SAS Display Manager erfolgt das Umschalten wie ich es oben mit den zwei Ziel-Zeilen des Windows-Start-Ikons gezeigt habe.
Dann kann Dein EG keine *falschen* Zeichen an das SAS auf dem Unix-System schicken.

Wie bringst Du den Code zum Ziel-System?

Gibt es eine Möglichkeit nach diesen Sonderzeichen gezielt zu suchen (..)
Mein Ansatz zur Code-Prüfung wäre:
Programm-Code als Text-Datei exportieren, dann einlesen in ein SAS Data Set (alles noch Unicode bzw. UTF-8).
Dann das SAS Data Set umsetzen von `outencoding="utf-8"` nach `outencoding="latin1"`:

```
libname rein_u8 "Quell_Pfad" outencoding="utf-8";
libname raus_l9 "Ziel_Pfad" outencoding="latin9";    /* auf Windows
data raus_l9.code_l9;
  set rein_u8.code_u8;    /* in code_u8 muss der Programmcode, einc
run;
```

Da der Code *Schrott*-Zeichen enthält, wird dieses Umsetzen abbrechen, das ist klar. ABER es bricht genau an der ersten Problem-Stelle ab! Das heißt das Ausgabe Data Set aus dem Umsetz-Data-Step enthält genauso so viele Sätze/Zeilen/Observations, wie SAS erfolgreich umsetzen bzw. schreiben konnte.
Damit kannst Du das 1. Problem beseitigen .....
Ich weiß, dass ist kein schöner Weg, aber es ist immerhin ein Weg.

Nächster Ansatz zur Code-Prüfung wäre:
Programm-Code als Text-Datei exportieren, dann einlesen in ein SAS Data Set (alles noch Unicode bzw. UTF-8).
Dann mit einem Data Step und translate alle *guten* Zeichen (a-z, A-Z, 0-9 etc pp) löschen.
Was übrig bleibt ist der *Schrott*, den man mit Edit->Find im Original-Programm-Code suchen und ändern kann ...
Auch kein schöner Weg, aber ...

Gruß
Hans Kneilmann, Schäfer Shop GmbH (SSI)

Danke!

17 January, 2014 - 09:19 — Student-Swen

Hallo Hans,

vielen Dank für die Hilfe!

ich konnte mit Deinen Tipps die problematischen Zeilen identifizieren. Habe dann die labels händisch neu geschrieben und ersetzt. Jetzt funktioniert es. Auch wenn ich optisch keinen Unterschied erkenne ;)

Viele Grüße

Swen

Log in or register to post comments

Super

17 January, 2014 - 12:21 — HansKneilmann

Hallo Swen,
prima, dass es geklappt hat.
Das mit dem *keinen Unterschied sehen* könnte an dem
Gedankenstrich und Bindestrich bzw. “en dash” und “em dash”
liegen. Auch die Gänsefüßchen (z.B. die eine Zeile hier drüber) sind nicht sofort als unterschiedlich zu erkenn, obwohl sie es sind.
Es gibt in den Zeichensätzen mittlerweile einiges an Stolpersteinen.

Gruß
Hans Kneilmann, Schäfer Shop GmbH (SSI)

Log in or register to post comments