Tipps und Tricks aus Gerds Fundgrube
UTF-8 in CSV-Dateien und das Problem mit Excel

Gerd Volberg, OPITZ CONSULTING Deutschland GmbH

Excel-Daten in einer Forms-Applikation zu erzeugen ist kein Hexenwerk. Dazu selektiert man in PL/SQL die benötigten Daten und speichert sie in einem CLOB als CSV ab. Dieses versendet man dann zum Beispiel als Dateianhang via E-Mail an den Anwender. Die Probleme fangen dann an, wenn Excel ins Spiel kommt.

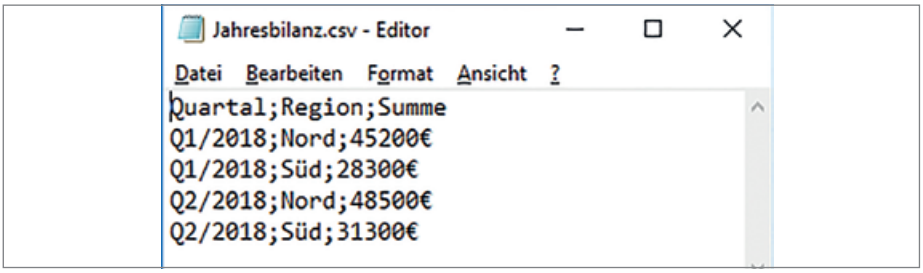

Abbildung 1: Ansicht der Rohdaten im Windows Editor

	A	B	C	D	E
1	Quartal	Region	Summe		
2	Q1/2018	Nord	45200â,¬		
3	Q1/2018	SÃ¼d	28300â,¬		
4	Q2/2018	Nord	48500â,¬		
5	Q2/2018	SÃ¼d	31300â,¬		
6					

Abbildung 2: Darstellung von UTF-8-Zeichen in Excel

Bytefolgen des BOM in verschiedenen Zeichenkodierungen

Kodierung	hexadezimale Darstellung	dezimale Darstellung
UTF-8	EF BB BF [4]	239 187 191

Abbildung 3: BOM-Artikel bei Wikipedia (wikipedia.org/wiki/Byte_Order_Mark)

	A	B	C	D	E
1	Quartal	Region	Summe		
2	Q1/2018	Nord	45.200 €		
3	Q1/2018	Süd	28.300 €		
4	Q2/2018	Nord	48.500 €		
5	Q2/2018	Süd	31.300 €		
6					

Abbildung 4: CSV-Datei mit BOM in Excel geöffnet

Jahresbilanz_mit_BOM.csv

```
Offset(h)  00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
00000000   EF BB BF 51 75 61 72 74 61 6C 3B 52 65 67 69 6F  ï»¿Quartal;Regio
00000010   6E 3B 53 75 6D 6D 65 0D 0A 51 31 2F 32 30 31 38  n;Summe..Q1/2018
```

Abbildung 5: CSV-Datei mit BOM im Hex-Editor

```
DECLARE
  V_UTF8_BOM   VARCHAR2 (10) := CHR (15711167);
  V_CLOB       CLOB;
BEGIN
...
  V_CLOB := V_UTF8_BOM || V_CLOB;
...
END;
```

Listing 1

Gehen wir einmal davon aus, dass die Datenbank einen UTF-8-Zeichensatz verwendet, die Forms-Applikation in diesem Zeichensatz kompiliert wurde und die CSV-Datei somit zu 100 Prozent aus UTF-8 besteht. Dies alles garantiert noch keine korrekte Darstellung der CSV-Daten in Excel.

Nehmen wir als Beispiel ein paar Bilanzdaten, die in unserer CSV-Datei gespeichert wurden. Der Windows-Editor hat sofort gemerkt, dass die Daten aus UTF-8 bestehen, und zeigt sie korrekt an *(siehe Abbildung 1)*. Wenn man das CSV per Doppelklick in Excel öffnet, sieht das hingegen weniger gut aus *(siehe Abbildung 2)*.

Deutsche Sonderzeichen werden nicht erkannt, das Euro-Symbol ist verschwunden, Eurowerte werden nicht als Zahlen erkannt und deswegen linksbündig dargestellt. So kann man nicht arbeiten.

Die Lösung

Byte Order Mark, auch „BOM" genannt, dient laut Wikipedia als Kennung zur Definition der Kodierungsfunktion in Unicode *(siehe Abbildung 3)*.

Lässt man eine Datei mit den Hex-Werten „EF BB BF" beginnen, werden die meisten Tools, wie auch Excel, erkennen, dass der weitere Text aus UTF-8 besteht. Das Ergebnis ist perfekt *(siehe Abbildung 4)*.

Die Änderungen am Source Code sind minimal. An der Stelle, an der man die fertige CSV-Datei versendet, muss vor dem Versand einfach nur der CLOB um das BOM erweitert werden *(siehe Listing 1)*. Die CSV-Datei sieht dann in einem Hex-Editor wie in *Abbildung 5* aus. Man erkennt sofort in den ersten drei Byte die Zeichenkette „EF BB BF":

Gerd Volberg
gerd.volberg@opitz-consulting.com
talk2gerd.blogspot.com