CSV-Datei per VBA öffnen
Versionen: Excel 2013, 2010, 2007, 2003 und 2002/XP

**FRAGE:** Wenn ich eine Datei mit der Endung .CSV manuell öffne, erkennt Excel automatisch die Feld-, Dezimal- und Tausender-Trennzeichen und teilt die Inhalte der CSV-Datei entsprechend auf mehrere Tabellenspalten auf. Wenn ich dieselbe Datei nun aber per VBA öffnen lasse – egal ob ich "Workbooks.Open" oder "Workbooks.OpenText" verwende – akzeptiert Excel die Feldtrennzeichen (Semikolons) in der Datei nicht und fügt die meisten Zeilen vollständig in die erste Tabellenspalte ein. Nur wenn Zahlenwerte mit Nachkommastellen vorhanden sind, trennt Excel die Daten unsinnigerweise an den Dezimalkommastellen. Erstaunlicherweise gelingt der Import mittels VBA-Makro fehlerfrei, wenn ich die Endung der Datei vorher von .CSV in .TXT ändere. Diesen Aufwand will ich bei der Makroentwicklung aber nicht betreiben. Kennen Sie vielleicht einen Trick, wie ich eine CSV-Datei ohne größeren Umstand per VBA öffnen kann?

**ANTWORT:** Excel entwickelt bei CSV-Dateien offenbar eine Eigenintelligenz, die dazu führt, dass ein korrektes Öffnen mittels VBA nur gelingt, wenn die CSV-Datei nach amerikanischem Vorbild aufgebaut ist: mit dem Komma als Feldtrennzeichen und dem Punkt als Dezimalkomma. Anders ist es nicht zu erklären, dass es keine Probleme gibt, wenn Sie einfach nur die Endung der Datei in .TXT ändern – wir können Ihre Beobachtung bestätigen.

Doch es gibt Abhilfe für deutsche Excel-Versionen und CSV-Dateien. Die "Open"- sowie die "OpenText"-Methoden des Workbook-Objekts verfügen über einen Parameter "Local". Wenn Sie "Local" auf den Wahrheitswert "True" setzen, verwendet Excel beim Einlesen der CSV-Datei die lokalen Systemeinstellungen. Mit folgendem Makro sollten Sie demnach keine Probleme haben, CSV-Dateien zu öffnen, die das Semikolon als Feldtrennzeichen benutzen:

Sub ImportCSV()
Dim strFilename As String

  strFilename = "C:\Daten\CSV\Test.csv"
  Application.Workbooks.OpenText \_
      Filename:=strFilename, \_
      Local:=True
End Sub

Sie müssen natürlich den Pfad- und Dateinamen in der Zeile "strFilename = ..." an Ihre Systemumgebung anpassen.