

SAP® Tabellen mit HTML lesen und anzeigen

SAP® Tabellen mit HTML lesen und anzeigen

von Stefan Schnell

Oftmals besteht die Notwendigkeit einfach nur bestimmte Daten eines SAP®-Systems anzuzeigen, und sei es schlicht aus informellen Gründen. Die Installation des SAP® GUI für Windows®, mit mehreren hundert Megabytes, scheint dafür etwas überdimensioniert zu sein. So bieten sich die SAP® Konnektoren NCo (dotNET) und JCo (Java) als schlanke Alternativen. Daneben existiert aber auch noch die Möglichkeit mit den SAP® ActiveX Steuerelementen auf ein SAP®-System zuzugreifen und solche Informationen aus Tabellen in Erfahrung zu bringen. In diesem Beitrag soll eine derartige Implementierung gezeigt werden. Da sich ActiveX Steuerelemente nahtlos in VBScript verwenden lassen, und VBScript nahtlos im Microsoft® Internet Explorer® verwendet werden kann, wird die Realisierung in diesem Umfeld vorgenommen.

Der Quellcode besteht aus zwei Blöcken. Der erste Block ist eine VBScript Sub-Prozedur und der zweite die GUI-Definition in HTML.

Ein wenig GUI

Beginnen wir mit der GUI-Definition. Sie ist sehr einfach gehalten und besteht aus einem Formular und einer Tabelle.

Im Formular werden u.a. die Verbindungsdaten wie Mandant, Benutzer usw. erfasst. Weiterhin der Tabellename, deren Daten angezeigt werden sollen, sowie ein Button zum Ausführen. Die folgende Tabelle dient als Container, in dem die Daten der SAP®-Tabelle angezeigt werden.

Das Lesen der Tabelle

Um die SAP®-Tabelle anzuzeigen werden drei Schritte verwendet. Der erste besteht darin die Verbindung zum SAP®-System aufzubauen. Besteht die Verbindung, so werden im zweiten Schritt, mit dem Funktionsbaustein RFC_READ_TABLE, aus der Tabelle DD03L die Kopfdaten der SAP®-Tabelle ermittelt, d.h. die Spaltennamen und deren Position. Diese werden dann nach ihrer Position sortiert und als erste Zeile in die Tabelle geschrieben. Im dritten Schritt werden jetzt die eigentlichen Tabellendaten, auch mit dem Funktionsbaustein RFC_READ_TABLE, ermittelt und als folgende Zeilen in der HTML-Tabelle angezeigt. Das Schreiben in die HTML-Tabelle wird über das Document Object Model (DOM) vorgenommen.

Einige Details

Zum Lesen der SAP®-Tabelle wird der Variablen ReadTableFunc der Funktionsbaustein RFC_READ_TABLE zugewiesen. Die Schnittstelle des Funktionsbausteines wird mit den Attributen Exports und Tables befüllt. Z.B. das Feldtrennzeichen DELEMETER mit ~ und der Tabellename QUERY_TABLE. Der Funktionsbaustein wird dann mit Call ausgeführt.

Das Ergebnis befindet sich in der Tabelle DATA im Feld WA. Sie wird Zeile für Zeile ausgelesen und über das DOM in die HTML-Tabelle geschrieben. Die Sortierung der Tabelle mit den Kopfdaten wird

mit einem Bubblesort-Algorithmus durchgeführt. Sie ist notwendig, da die Daten in beliebiger Reihenfolge zurückgeliefert werden können. Dies wäre jedoch sehr verwirrend.

Zur Beachtung

Der gezeigte Ansatz funktioniert nur mit dem Microsoft® Internet Explorer®, da nur dieser den Einsatz von ActiveX Steuerelementen erlaubt. Es sollte auf jeden Fall darauf geachtet werden, dass die Sicherheitseinstellungen des Internet Explorer® die Ausführung von aktiven Inhalten gestattet. Alternativ kann auch die Extension der Datei von html in hta (HTML Application) umbenannt werden.

Fazit

Das Lesen von SAP®-Tabellen und das Anzeigen der Tabelleninhalte innerhalb einer HTML-Datei mit VBScript und den SAP® ActiveX Steuerelementen bietet weitere interessante Möglichkeiten, neben den Standardkonnektoren. Damit ist es relativ einfach möglich Informationen zu erhalten und diese mit HTML und CSS visuell ansprechend aufzubereiten. Auch die notwendige Software-Basis kann sehr klein gehalten werden resp. besteht aus Standards des Betriebssystems Microsoft® Windows®.

Haben Sie Fragen, Anregungen oder Anmerkungen, so senden Sie einfach eine E-Mail an mail@stschnell.de.

Warenzeichen

- SAP ist ein eingetragenes Warenzeichen der SAP AG
- Microsoft, Internet Explorer, Visual Basic Scripting und ActiveX sind eingetragene Warenzeichen der Microsoft Corporation

Links

- www.stschnell.de

SAP® Tabellen mit HTML lesen und anzeigen

```

<!-- Begin-----
This HTML and VBScript source shows how to connect a SAP system via
SAP ActiveX control libraries, to read data from a SAP table and to
view this data in a HTML table, with header line.

Author:  Stefan Schnell
Version: 0.61

----- -->

<html>

<head>

<title>
  Connect a SAP system and view table content
</title>

<hta:application applicationname="ReadSAPTable" id="ReadSAPTable"
  version="1.0"/>

<!-- To encode your source, use the Script Encoder screnc.exe -->
<script language="VBScript.Encode">

  '-Directives-----
    Option Explicit

  '-Sub GetData-----
    Sub GetData()

      '-Variables-----
        Dim SAPFunc, Connection, SAPConnection, ReadTableFunc
        Dim Param, Table, RowCount, i, j, DataLine, Node, tmp
        Dim Tabelle, TableName, tr, td, arrDataLine, arrFields
        Dim fieldName

      '-Get the name of the table-----
        TableName = CStr(Document.eingabe.TableName.Value)
        If Trim(TableName) = "" Then
          MsgBox "No SAP table name specify", vbOKOnly, "Error"
          Exit Sub
        End If

      '-Get SAP.Functions-----
      '-
      '- If it is not possible to create SAP.Functions instance,
      '- Microsoft(c) Internet Explorer(c) creates an error
      '-
      '------
        Set SAPFunc = CreateObject("SAP.Functions")
        If Not IsObject(SAPFunc) Then
          Exit Sub
        End If

      '-Get SAP.LogonControl connection-----
        Set Connection = SAPFunc.Connection()
        If Not IsObject(Connection) Then
          MsgBox "No SAP.Connection instance", vbOKOnly, "Error"
          Exit Sub
        End If
    End Sub
  End Sub
</script>

```

SAP® Tabellen mit HTML lesen und anzeigen

```

'-Set connection parameters-----
Connection.Client = CStr(Document.eingabe.Client.Value)
Connection.User = CStr(Document.eingabe.User.Value)
Connection.Password = CStr(Document.eingabe.Password.Value)
Connection.Language = "DE"
Connection.System = CStr(Document.eingabe.System.Value)
Connection.HostName = CStr(Document.eingabe.HostName.Value)
Connection.SystemNumber = _
    CStr(Document.eingabe.SystemNumber.Value)

'-Connect SAP system-----
SAPConnection = Connection.Logon(0, vbTrue)

If SAPConnection = vbTrue Then

    '-Get function module RFC_READ_TABLE-----
    Set ReadTableFunc = SAPFunc.Add("RFC_READ_TABLE")

    If IsObject(ReadTableFunc) Then

        '-Clear HTML table-----
        Set Tabelle = Document.getElementById("Tabelle")
        j = Tabelle.childNodes.length
        If j > 0 Then
            For i = 1 To j
                Set Node = Tabelle.lastChild
                Tabelle.removeChild(Node)
                Set Node = Nothing
            Next
        End If

        '-Define export parameter DELIMITER to ~-----
        Set Param = ReadTableFunc.Exports("DELIMITER")
        Param.Value = "~"

        '-Get header line informations from the SAP table-----
        '-Define export parameter QUERY_TABLE-----
        Set Param = ReadTableFunc.Exports("QUERY_TABLE")
        Param.Value = "DD03L"
        '-Define export parameter OPTIONS-----
        Set Param = ReadTableFunc.Tables("OPTIONS").Rows.Add
        Param("TEXT") = "TABNAME = '" & TableName & "'"
        '-Define export parameter FIELDS-----
        Set Param = ReadTableFunc.Tables("FIELDS").Rows.Add
        Param("FIELDNAME") = "POSITION"
        Set Param = ReadTableFunc.Tables("FIELDS").Rows.Add
        Param("FIELDNAME") = "FIELDNAME"
        '-Read table-----
        If ReadTableFunc.Call() = vbTrue Then
            Set Table = ReadTableFunc.Tables("DATA")
            If IsObject(Table) Then
                RowCount = Table.RowCount()
                '-Copy result to an array-----
                ReDim arrFields(RowCount - 1)
                For i = 0 To RowCount - 1
                    arrFields(i) = Table.Value(i + 1, "WA")
                Next
                ReadTableFunc.Tables("DATA").Rows.RemoveAll
                Set Table = Nothing
            Else
                MsgBox "No SAP table instance", vbOKOnly, "Error"
                Exit Sub
            End If
        End If
    End If
End If

```

SAP® Tabellen mit HTML lesen und anzeigen

```

    End If
Else
    MsgBox "RFC_READ_TABLE DD03L not successful", _
        vbOKOnly, "Error"
    Exit Sub
End If
'-Sort array with field names (BubbleSort)-----
For i = 0 To UBound(arrFields)
    For j = i + 1 To UBound(arrFields)
        If arrFields(i) > arrFields(j) Then
            tmp = arrFields(i)
            arrFields(i) = arrFields(j)
            arrFields(j) = tmp
        End If
    Next
Next
'-Print field names in the HTML table-----
If UBound(arrFields) > 0 Then
    Set tr = Tabelle.insertRow(0)
    For j = 0 To UBound(arrFields)
        Set td = tr.insertCell()
        FieldName = Split(arrFields(j), "~")
        td.innerText = FieldName(1)
    Next
Else
    MsgBox "No head line", vbOKOnly, "Information"
End If

'-Get SAP table data-----
'-Define export parameter QUERY_TABLE-----
Set Param = ReadTableFunc.Exports("QUERY_TABLE")
Param.Value = TableName
'-Delete entries-----
ReadTableFunc.Tables("OPTIONS").Rows.RemoveAll
ReadTableFunc.Tables("FIELDS").Rows.RemoveAll
'-Read table-----
If ReadTableFunc.Call() = vbTrue Then
    Set Table = ReadTableFunc.Tables("DATA")
    If IsObject(Table) Then
        RowCount = Table.RowCount()
        '-Print SAP table content to HTML table-----
        If RowCount > 0 Then
            For i = 1 To RowCount
                Set DataLine = _
                    Document.createTextNode(Table.Value(i, "WA"))
                arrDataLine = Split(DataLine.data, "~")
                Set tr = Tabelle.insertRow(i)
                For j = 0 To UBound(arrDataLine)
                    Set td = tr.insertCell()
                    If Trim(arrDataLine(j)) = "" Then
                        td.innerHTML = "&nbsp;"
                    Else
                        td.innerText = arrDataLine(j)
                    End If
                Next
            Next
        Else
            MsgBox "No data", vbOKOnly, "Information"
        End If
        Set Table = Nothing
    Else
        MsgBox "No SAP table instance", vbOKOnly, "Error"
    End If
End If

```

SAP® Tabellen mit HTML lesen und anzeigen

```

        End If
    Else
        MsgBox "RFC_READ_TABLE " & TableName & " not successful", _
            vbOKOnly, "Error"
    End If

    Set td = Nothing
    Set tr = Nothing
    Set Tabelle = Nothing
    Set ReadTableFunc = Nothing

Else
    MsgBox "No RFC_READ_TABLE instance", vbOKOnly, "Fehler"
End If

'--Logoff-----
    Connection.Logoff()

Else
    MsgBox "No connection to SAP system", vbOKOnly, "Error"
End If

End Sub

</script>

</head>

<!-- GUI----- -->

<body>

<h2 style="font-family:Arial;">
    Connect a SAP system and view table content
</h2>

<form name="eingabe" style="font-family:Arial;">
    <!-- Client (Mandant) -->
    Client:&nbsp;<input type="text" name="Client" size="3" value="000">
    &nbsp;&nbsp;&nbsp;
    <!-- User (Benutzer) -->
    User:&nbsp;<input type="text" name="User" size="15" value="BCUSER">
    &nbsp;&nbsp;&nbsp;
    <!-- Password -->
    Password:&nbsp;<input type="password" name="Password" size="25"
        value="minisap">
    <br />
    <!-- System (SID) -->
    System:&nbsp;<input type="text" name="System" size="3" value="NSP">
    &nbsp;&nbsp;&nbsp;
    <!-- Hostname -->
    Hostname:&nbsp;<input type="text" name="HostName" size="25"
        value="192.168.28.139">
    &nbsp;&nbsp;&nbsp;
    <!-- Systemnumber -->
    Systemnumber:&nbsp;<input type="text" name="SystemNumber" size="2"
        value="00">
    <br /><br />
    <!-- Tablename -->
    Table name:&nbsp;<input type="text" name="TableName" size="30">
    &nbsp;&nbsp;&nbsp;

```

SAP® Tabellen mit HTML lesen und anzeigen

```

    <!-- Button to view the table -->
    <input type="button" value="GetData" onClick='GetData()' '>
</form>

<table id="Tabelle" border="1">
</table>

</body>

</html>

<!-- End----->

```

MANDT	CARRID	CONNID	FLDATE	PRICE	CURRENCY	PLANETYPE	SEATSMAX	SEATSOCC	PAY
000	AA	0017	20080130	422.94	USD	747-400	385	371	
000	AA	0017	20080227	422.94	USD	747-400	385	372	
000	AA	0017	20080326	422.94	USD	747-400	385	374	
000	AA	0017	20080423	422.94	USD	747-400	385	371	
000	AA	0017	20080521	422.94	USD	747-400	385	373	
000	AA	0017	20080618		USD	747-400	385	370	