

Webkartei

Die Zeiten vor den mächtigen Frameworks

Die Beschreibung „Webkartei“ bezieht sich auf die kleinen Programme, mit deren Hilfe die Karteikarten der Projektseiten *Periodensystem der Elemente*, *Heilpflanzen-Kompendium* und *Ornithologie-Kompendium* erstellt wurden. In dieser Kurzbeschreibung wird auch die Verwendung externer Dateien als Datenquellen beschrieben, allerdings nicht unter Verwendung von JavaScript. Informationen zu den vorgenannten Webseiten finden Sie in meinem Repositorium *Historische Webseiten*. Die folgende Beschreibung stammt übrigens aus dem Jahr 1999.

Die Darstellung von Informationen auf virtuellen Karteikarten hat sicher gewisse Vorzüge, denn es werden jeweils nur bestimmte Informationsausschnitte recht überschaubar angeboten, wobei der Abruf weiterer Informationen über die Register der Kartei möglich ist. Da die gesamte Präsentation jedoch im HTML-Format abgebildet war, welches normalerweise keine dynamischen Elemente kennt, musste eine Struktur definiert werden, in der wenigstens die Daten dynamisch vorgehalten werden konnten.

Hierbei besteht eine Karteikarte im günstigsten Fall aus nur einem Register; allerdings braucht man hier schon wenigstens 2 Dateien, um diese Karte abzubilden, nämlich die HTML-Datei und die Datei, in der die Script-Variablen stehen. Auf die Verwendung von Script-Variablen wird hier jedoch nicht näher eingegangen. Mit einer anderen Darstellungsweise, z. B. in PHP oder JAVA, hätte man nur einmal ein Anzeigeelement definieren müssen, das dann jeweils mit den Daten aus einer zentralen Datenquelle, z. B. aus einer Tabelle, angezeigt werden könnte. Webspace-Restriktionen ließen jedoch eine solche Darstellung damals leider nicht zu, so dass die dynamische und interaktive Darstellung nur in der vorgestellten Weise erreicht werden konnte.

Ohne jetzt allzu tief in die Programmierung mit Q-BASIC einzusteigen, brauchen Sie eigentlich nur einige Grundkenntnisse, um ein solches Karteikartensystem zu erstellen. Zunächst entwerfen Sie die Karteikarte als reines HTML-Dokument. Dort, wo später die variablen Informationen angezeigt werden sollen, setzen Sie einfach Platzhalter (Punkte oder Variablennamen etc.). Die variablen Daten, die in Tabellen aufbereitet und als CSV-Datei gespeichert wurden, stehen in einer für Q-BASIC leicht nutzbaren Form also zur Verfügung. Beispieltabelle (CSV) für Daten:

```
1;Schwarzer Holunder;Eller, Holder, Huller;Sambucus nigra;...  
2;Pfefferminze;Aderminze, Edelminze;Mentha piperita;...  
3;Arnika;Bergwohlverleih;Arnica montana;...
```

Das Grundgerüst für die Kartei, die im folgenden Beispiel aus nur zwei Registern besteht, macht jedoch deutlich, dass bei HTML-Darstellung durch Anklicken des Links für die weitere Karte des Karteisystems wieder eine neue HTML-Seite geöffnet werden muss. Dabei ist jeweils zu beachten, dass sich beide Karten der angezeigten Kartei auf ein und dasselbe Element beziehen. Das muss dann auch in dem zu erstellenden Q-BASIC-Programm berücksichtigt werden, d. h. also, die Dateiaufrufe müssen im Programm variabel gehalten werden. Eine Karteikarte kann folgendes Aussehen haben:



Durch Anklicken der Karteikarten (Register) wird die jeweilige Seite mit den dazugehörigen Informationen angezeigt. Die oben dargestellte Karte sieht in der HTML-Codierung wie folgt aus:

```
<table border="0" width="60%"><tr>
<td bgcolor="#AABBE" width="20%"><font face="Arial, Sans-Serif"
size="2">Karte A</font></td>
<td bgcolor="#AABBE" width="20%"><font face="Arial, Sans-Serif"
size="2">Karte B</font></td>
<td bgcolor="#FFFFFF" width="60%"><font face="Arial, Sans-Serif" size="2">
</font></td>
</tr></table>
<table border="0" width="60%"><tr><td bgcolor="#DDDDDD" width="100%"
height="170"
valign="top">
<font face="Arial, Sans-Serif" size="2"><br>Information 1<br>
Information 2<br>Information 3<br></font></td></tr></table>
```

In Q-BASIC muss nun per Printbefehl die entsprechende Kartei geschrieben werden. Dabei ist zu beachten, dass die Anführungszeichen in Q-BASIC codiert werden müssen, da auch der Printbefehl diese Anführungszeichen verwendet. Jedes Anführungszeichen muss in der Zeichenkette des Printbefehls mit **CHR\$(34)** übersetzt werden. Im obigen Beispiel für die Karteikarte ist jedoch noch nicht berücksichtigt, dass beim Anklicken auf **Karte B** die neue Datei angezeigt wird (Referenzierung, Link). Dieser Link muss zur Hauptkarte passen. Nehmen wir an, es sollen 10 Karteien erstellt werden, mit denen die 10 Einträge aus einer CSV-Datei auf Karteikarten abgebildet werden sollen, so kann der BASIC-Code folgendermaßen aussehen:

```
CLS
PRINT "Starten der Bearbeitung ..."
OPEN "daten.csv" FOR INPUT AS #1
DO UNTIL (EOF(1))
  LINE INPUT #1, Zeile$
  k% = k% + 1
  FOR i% = 1 TO LEN(Zeile$)
    IF MID$(Zeile$, i%, 1) = ";" THEN
      z% = z% + 1
      IF z% = 1 THEN s1% = i%
      IF z% = 2 THEN s2% = i%
      IF z% = 3 THEN s3% = i%
      IF z% = 4 THEN s4% = i%
      IF z% = 5 THEN s5% = i%
      IF z% = 6 THEN s6% = i%
      IF z% = 7 THEN s7% = i%
      IF z% = 8 THEN s8% = i%
      IF z% = 9 THEN s9% = i%
      IF z% = 10 THEN s10% = i%
```

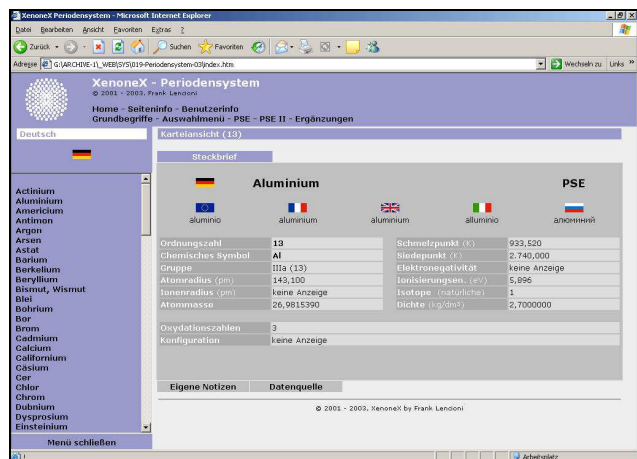
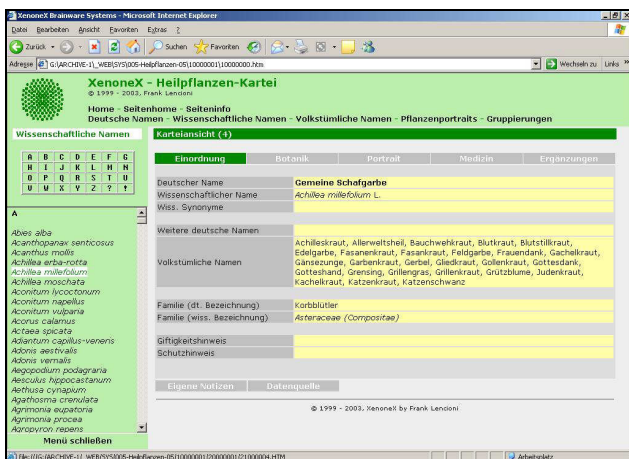
```

END IF
NEXT i%
Z% = 0
v1$ = MID$(Zeile$, 1, (s1% - 1))
v2$ = MID$(Zeile$, (s1% + 1), (s2% - s1% - 1))
v3$ = MID$(Zeile$, (s2% + 1), (s3% - s2% - 1))
v4$ = MID$(Zeile$, (s3% + 1), (s4% - s3% - 1))
v5$ = MID$(Zeile$, (s4% + 1), (s5% - s4% - 1))
v6$ = MID$(Zeile$, (s5% + 1), (s6% - s5% - 1))
v7$ = MID$(Zeile$, (s6% + 1), (s7% - s6% - 1))
v8$ = MID$(Zeile$, (s7% + 1), (s8% - s7% - 1))
v9$ = MID$(Zeile$, (s8% + 1), (s9% - s8% - 1))
v10$ = MID$(Zeile$, (s9% + 1), (s10% - s9% - 1))
IF k% < 10 THEN Ausgabe$ = "KarteA0" + RIGHT$(STR$(k%), 1) + ".HTM"
IF k% >= 10 THEN Ausgabe$ = "KarteA" + RIGHT$(STR$(k%), 2) + ".HTM"
OPEN Ausgabe$ FOR APPEND AS #2
PRINT #2, "<table border=" + CHR$(34) + "0" + CHR$(34) +
"width=" + CHR$(34) + ...
:
:
PRINT #2, v2$ + "<br>" + v3$ + "<br></td></tr></table>"
CLOSE #2
CLOSE #1
PRINT "Fertig!"
END

```

Damit die Anzeige nicht noch unübersichtlicher wird, wurden einige Einträge, die zur Darstellung der HTML-Codes dienen, weggelassen. Bei den Projektseiten fanden Sie früher die Downloads der fertig bearbeiteten Projekte - und dort auch die Programme, mit deren Hilfe diese Systeme erstellt wurden. Dort konnten Sie sich die Quellen der kleinen Programme anschauen, die damals dort natürlich vollständig abgebildet waren. *[Diese alte Seite gibt es heute natürlich nicht mehr - ebenso wenig wie die angesprochenen Downloads.]*

Das obige Programm schreibt erst einmal nur die HTML-Dokumente für die **Kartei A**, die in den Namen *KarteiA01.HTM* bis *KarteiA10.HTM* durchnummeriert ist. Die Aufrufe für **Kartei B** (mit den Namen *KarteiB01.HTM* bis *KarteiB10.HTM*) sind hier erst einmal nicht berücksichtigt, können aber aufgrund der Zählvariable **k%** ebenso leicht eingebaut werden. Wenn Sie nun die Variablen, die auf Kartei B angezeigt werden sollen, im Quelltext ändern, und auch den Dateinamen anpassen, haben Sie im Nu die gewünschten 20 Karten (= 20 Dateien) erstellt. Eine kleine Anzahl Karteien lässt sich sicherlich auch noch mit *Copy and Paste* erstellen, aber versuchen Sie das einmal mit 477 Heilpflanzen, wobei jede Kartei 5 Register hatte. Hier ist leicht zu erkennen, dass man ein solches System, in dem dann alleine zur Karteidarstellung 2.385 Dateien geschrieben werden müssen, nicht mehr von Hand zu Fuß erstellen kann.



Mit dieser Kurzdarstellung habe ich Ihnen nur zeigen wollen, wie man - trotz (oder gerade wegen) aller Webspace-Einschränkungen - Seitensysteme erstellen kann, mit denen eine Fülle von Informationen auf simpelste Weise in Karteisystemen unterzubringen ist, die man lokal sogar dynamisch halten kann. Ohne diese Idee (und deren Realisierung) wäre die dynamische Präsentation des Heilpflanzen-Kompendiums, das in der Endfassung etwa 28 MB groß war, überhaupt nicht möglich gewesen. Auch wenn dort noch längst nicht alle Daten verfügbar waren, die in reiner Fleißarbeit hätten nachgetragen werden müssen, so ist das System selbst sicherlich eine Meisterleistung in puncto dynamischer Datenorganisation und Koordination mit einfachsten Mitteln gewesen.

[Mit 28 MB und mehreren tausend Dateien stieß ich seinerzeit an die Grenzen dessen, was mein kostenloser Hoster gerade so zuließ. Heute arbeite ich in solchen Fällen mit Template-Bausteinen und bei einem solchen Umfang sinnvollerweise mit Datenbanken. Und auch die einfache Karteiabildung würde ich heute eher als DOM-Element generieren. Frank Lencioni]

Verwendungshinweis

Die unveränderte Weitergabe dieses Dokumentes ist gestattet. Auch die Verwendung einzelner Passagen oder Graphiken aus diesem Dokument ist unter Berücksichtigung der Quellenangabe erlaubt. Nicht gestattet ist die Verbreitung des Dokumentes oder einzelner Passagen bzw. Graphiken aus diesem Dokument im Zusammenhang mit Inhalten, Darstellungen, Aufforderungen und Äußerungen sexistischer, rassistischer, volksverhetzender, religiös diskriminierender oder sonst gesetzeswidriger Ausprägung. *Frank Lencioni*

E-Mail: Frank@Lencioni.de / Web: <http://Frank.Lencioni.de>