

WEB-Publishing und SAS®-Anwendungen auf dem Application Server mit HTMLOUT

Wilfried Schollenberger

WS Unternehmensberatung und Controlling Systeme GmbH, Heidelberg

Abstract

HTML bewährt sich als u.a. als Mittel zur Veröffentlichung von Dokumenten und Daten im Internet und in Intranets. SAS Institute stellt mit den WEB-Publishing Tools und der SAS/IntrNet Software die erforderliche Umgebung für statisches Reporting und dynamische Auswertungen, die zum Zeitpunkt der Abfrage durchgeführt werden, bereit.

Speziell für Anwendungen auf dem Application Server und für die automatisierte Erstellung verknüpfter HTML-Seiten beim WEB-Publishing hat die WS GmbH das Ausgabe-Tool HTMLOUT entwickelt. Durch die Trennung der in HTML implementierten Darstellung von der in SAS implementierten Datenbereitstellung, wird die Entwicklung wesentlich vereinfacht und beschleunigt.

Inhalt

1. HTML als Standard für Informationen und Frontend für Applikationen
2. WEB-Publishing
 - 2.1 Beispiel: Dokumentation von Datenbanken
 - 2.2 Beispiel: Publikation von statistischen Übersichten
 - 2.3 JavaScript und Java Applets ?
3. Dynamische Abfragen mit dem Application Server
 - 3.1 Beispiel: Korrelationskoeffizienten
 - 3.2 Beispiel: Projektmanagement
4. Zusammenfassung

1. HTML als Standard für Informationen und Frontend für Applikationen

Ursprünglich als Medium zur internationalen Kommunikation von Wissenschaftlern konzipiert, haben sich das WEB und HTML inzwischen zu allgemein genutzten Medien für viele Arten von Informationen entwickelt. Insbesondere die unternehmens- und organisationsinterne Verbreitung von Dokumenten wurde wesentlich vereinfacht, weil die vereinzelt Dateien auf Servern durch ein strukturiertes System von Dokumenten ersetzt wurden. Diese Dokumente werden mit Hyperlinks verknüpft, so daß der Leser Verweise schnell nachvollziehen kann.

HTML bewährt sich aber auch als Frontend für Informationssysteme, die ihre Ergebnisse in diesem Format ausgeben oder auf einem Server laufen und direkt vom Browser des Benutzers aufgerufen werden. Damit lassen sich statisches Reporting (als Ersatz der bisherigen Listen) und ad hoc Abfragen nahtlos unter einer Oberfläche integrieren und einem wesentlich größeren Adressatenkreis zugänglich machen. Quasi nebenbei entfällt damit auch die Installation und Pflege von Programmen auf den Benutzer-PC's und das Aufrüsten der Hardware für die erhöhten Anforderungen der jeweils nächsten Version.

2. WEB-Publishing

Genaugenommen versteht man unter WEB-publishing die Erstellung von Dokumenten in genormten, hersteller- und betriebssystemunabhängigen Formaten, wie HTML für Texte, GIF für Grafiken und JPEG für Bilder, die über einen HTTP-Server verbreitet und von einem Browser angefordert werden. Nahezu alle Softwarepakete können inzwischen diese Formate

erstellen. Von SAS Institute sind die WEB-Publishing-Tools über das Internet kostenlos erhältlich. Damit lassen sich u.a. SAS-Output in HTML-Seiten und Proc Tabulate Ausgaben in HTML-Tabellen umwandeln, sowie SAS-Dateien in HTML-Tabellen ausgeben. Darüberhinaus lassen sich mit SAS/Graph GIF- und JPEG-Dateien erstellen.

Komplexere Ausgaben mit automatisch eingebundenen Links erfordern jedoch entweder eine manuelle Nachbearbeitung oder eine eigene Programmierung. Z.B. muß irgendwie angegeben werden, wie die Adresse der Verknüpfung aus den Daten ermittelt werden kann. Ebenso sind alle Ausgaben, die bisher in Datasteps programmiert wurden, natürlich auch weiterhin nur realisierbar, wenn der Entwickler die Ausgabe genau gestalten kann. Möglich ist das z.B., indem die HTML-Elemente in das Datastep-Programm mit eingebaut werden. Einfacher ist es jedoch, wenn man den HTML-Teil von der SAS-Programmierung getrennt entwickeln kann, und die Daten zur Laufzeit an die entsprechenden Stellen gesetzt werden. Hierzu hat die WS GmbH das Tool HTMLOUT entwickelt: Zunächst wird ein sogenanntes HTML-Template wie eine ganz normale HTML-Seite entwickelt. Die Stellen, an denen später die Daten stehen sollen, werden durch Platzhalter markiert. Dann werden in HTML-Kommentaren die Anweisungen für HTMLOUT eingefügt. Darin steht z.B. aus welcher SAS-Datei gelesen werden soll, und daß ein bestimmter Abschnitt nur unter bestimmten Bedingungen ausgegeben wird.

Der größte Vorteil gegenüber der Programmierung in einem Datastep ist, daß sowohl die Erstellung des HTML-Template wie auch die Datenbereitstellung in SAS zu relativ einfachen, leicht und schnell zu lösenden Aufgaben werden. Andererseits ist diese Kombination wesentlich mächtiger als "WEB-to-Database"-Tools, denen es oft an den notwendigen Features zur Datenaufbereitung fehlt. (Bei online-Abfragen kommen noch weitere Einschränkungen hinzu, sie den Abschnitt zum Application Server).

Alle folgenden Beispiele sind mit HTMLOUT erstellt worden.

2.1 Beispiel: Dokumentation von Datenbanken

Das erste Beispiel ist ein Modellierungs- und Dokumentationstool für Informationsdatenbanken (Enterprise Data Warehouse). Die fachlichen Anforderungen und allgemeine Beschreibungen werden über einen HTML-Editor, z.B. Netscape Composer, erstellt. Die Entity-Relationship- und die Datenflußdiagramme werden mit einem Flowcharter erstellt und als GIF-Dateien exportiert. Der Aufbau der Datenbank und die genauen Definitionen für die einzelnen Elemente werden über ein Tool in einer Datenbank erfaßt. Aus diesen Informationen wird dann automatisch in ein System verknüpfter HTML-Dateien erstellt, das z.B. über einen Intranet-Server allen berechtigten Interessenten zugänglich gemacht wird.

Auf diese Art sind die Informationen wesentlich schneller zu finden als in Handbüchern und Aktenordnern. Darüberhinaus verhindert die zentrale elektronische Speicherung, daß gedruckte Dokumente mangels Aktualisierungen veralten.

2.2 Beispiel: Publikation von statistischen Übersichten

Häufig muß bei gedruckten Publikationen die detaillierte Darstellung der Daten aus Platzgründen unterbleiben. Bestenfalls werden solche Ergebnisse in einem Materialien-Anhang zusammengefaßt, wo sie oft nur schwer nach einigem Blättern zu finden sind.

Mit HTML lassen sich dagegen solche Zusatzinformationen über Verweise direkt anspringen, ohne den eigentlichen Inhalt zu überlagern. Ein Beispiel ist diese Liste mit statischen Verteilungsparametern, wo zu jeder Variablen auch eine Balkengrafik abgerufen werden kann.

Tabelle 1: Statistiken für Datei sasuser.fitness

Name	Label	N	Mittelwert	Stdev	Min	Max
<u>AGE</u>	Age in years	31	47.68	5.21	38.00	57.00
<u>WEIGHT</u>	Weight in kg	31	77.44	8.33	59.08	91.63
<u>RUNTIME</u>	Min. to run 1.5 miles	31	10.59	1.39	8.17	14.03
<u>RSTPULSE</u>	Heart rate while resting	31	53.74	8.29	40.00	76.00
<u>RUNPULSE</u>	Heart rate while running	31	169.65	10.25	146.00	186.00
<u>MAXPULSE</u>	Maximum heart rate	31	173.77	9.16	155.00	192.00
<u>OXYGEN</u>	Oxygen consumption	31	47.38	5.33	37.39	60.06
<u>GROUP</u>	Experimental group	31	1.03	0.84	0.00	2.00

Die Erstellung einer solchen Tabelle und der entsprechenden Grafiken kann mit einem einfachen Macro-Aufruf automatisiert werden:

```
%wsg_sbar(sasuser.fitness,          /* Dateiname */
  var      = _numeric_ ,          /* Variablen */
  prefix   = .\ch_Fitness_ ,      /* Prefix für Namen der Grafiken */
  dsstats  = work.ergstats,      /* Dateiname für Zwischenergebnisse */
  html     = Stats1.html,        /* Name des HTML-Templates */
  chartxy  = 500x400             /* Format der Grafiken in Pixel */
);
```

Das Template zu der obigen Ausgabe hat folgenden Aufbau:

Statistiken für Datei #?&dsname;

Name	Label	N	Mittelwert	Stdev	Min	Max
#varname;	#varLabel;	#N:8.0;	#MEAN:8.2;	#Std:8.2;	#Min:8.2;	#Max:8.2;

2.3 JavaScript und Java Applets?

In meinen älteren Vorträgen lautete die klare Empfehlung, auf nicht genormte Erweiterungen wie Frames, JavaScript und Java zu verzichten, es sei denn ihre Verwendung sei aufgrund besonderer Anforderungen unbedingt notwendig. Im Internet gilt noch immer, daß insbesondere JavaScript und Java nicht vorausgesetzt werden können - auch weil viele diese Features aufgrund von Sicherheitsbedenken und zur Umgehung von Performance-Problemen deaktivieren. Bei Frames empfinde ich es ärgerlich, daß sich eine bestimmte Stelle nicht direkt als Bookmark (Lesezeichen) aufnehmen läßt, was aber die meisten anderen nicht zu stören scheint. Andere Kritikpunkte waren die rasche Entwicklung der verschiedenen Java-Versionen, die zu einem schnellen Veralten des Codes führt, und Probleme beim Drucken.

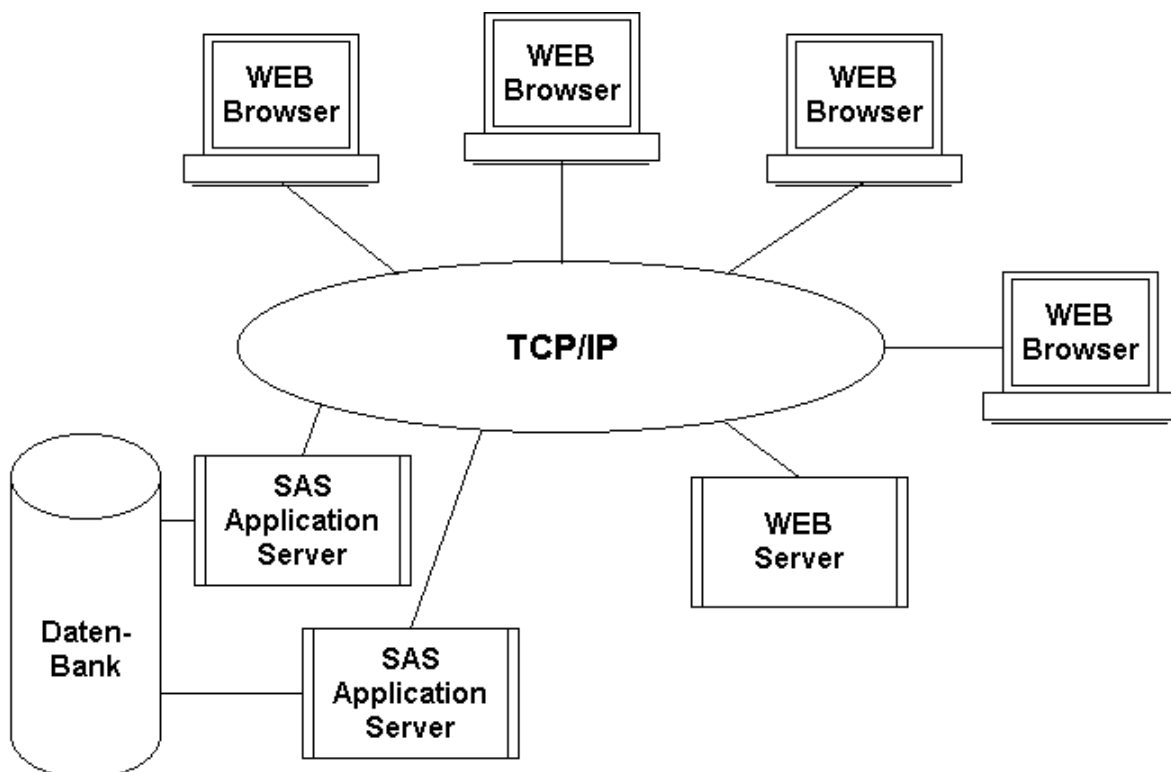
Unternehmensintern kann die Situation natürlich auch ganz anders aussehen, z.B. wenn schon in anderen Projekten Java-Knowhow aufgebaut wird. Insbesondere bei dynamischen Anwendungen mit dem Application Server lassen sich einige Funktionen mit HTML alleine gar nicht realisieren. Wenn z.B. ein "Zurück"-Button eine oder mehrere Seiten überspringen soll, geht das nur mit JavaScript. Mit sinnvollen Java Applets läßt sich der Server entlasten.

Ein gutes Beispiel für den effizienten Einsatz eines Java Applets ist die gemischte Darstellung von Text und Grafik. Wenn für die Grafik ein Java Applet verwendet wird, reduziert sich der dynamische Teil auf eine einfache HTML-Seite, in der auch die Parameter für das Applet übergeben werden. Andernfalls hätte zusätzlich zur HTML-Seite auch noch eine GIF-Datei erstellt und übertragen werden müssen.

3. Dynamische Abfragen mit dem Application Server

Mit der SAS/IntrNet® Software lassen sich solche Ausgaben auch ad hoc zur Laufzeit erstellen. Der einzige zusätzliche Aufwand für den Entwickler besteht im Aufruf, der in eine HTML-Seite eingebaut werden muß, und in kleinen SCL-Programmen, die diese Aufrufe verarbeiten.

Von den verschiedenen Optionen, die SAS Institute mit der SAS/IntrNet Software anbietet, wird für den schnellen produktiven Einsatz der SAS Application Server mit HTML 3.2 als Frontend empfohlen. In dieser Konfiguration laufen mehrere SAS-Sessions (auf einem oder mehreren Rechnern) und warten auf Anforderungen.



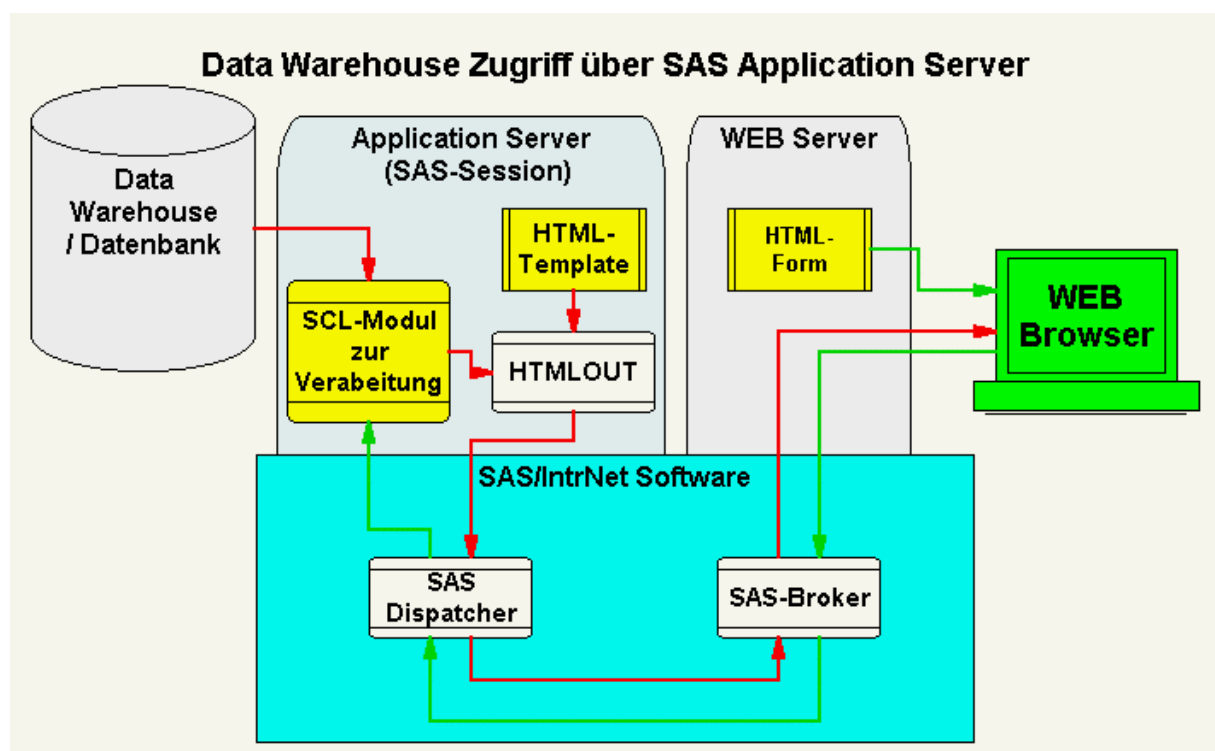
Die Benutzer starten an Ihrem Browser eine Anforderung an einen HTTP-Server (wie in der Grafik auch oft WEB-Server genannt). Dieser HTTP-Server gibt die Anforderung über einen CGI-Aufruf der "Broker"-Komponente von SAS/IntrNet an eine freie SAS-Session weiter, die die Anforderung bearbeitet und das Ergebnis über den HTTP-Server an den Browser des Benutzers zurückgibt. In der Regel erfolgt die Ausgabe im HTML-Format, es sind aber je nach Anwendung auch andere Formate, z.B. GIF für Grafiken oder tab-separated-values zur Weiterverarbeitung in einem Spreadsheet möglich.

Diese Konfiguration hat gleich mehrere Vorteile:

- Sie ist einfach skalierbar, indem zusätzliche oder leistungsfähigere Server eingebunden werden.
- Die Anforderungen an die Endgeräte der Benutzer sind gering: keine spezielle Java-Version, keine besonderen Anforderungen an die Leistung.

- Das System ist robust gegen Überlast, der Benutzer erhält die Meldung, daß gerade kein Server frei ist, aber die Server selbst werden nicht überlastet, d.h. es kann nicht zu der Situation kommen, daß sich zuviele Abfragen gegenseitig blockieren.
- HTML ist durch das WWW weit verbreitet und die Benutzer verstehen das Interface ohne zusätzliche Schulungen. Natürlich müssen die Seiten entsprechende Erläuterungen enthalten.

Serverseitig wird empfohlen, grundsätzlich nur SCL-Entries aufrufen. Diese Variante ist die mächtigste und ermöglicht auch anspruchsvolle Algorithmen, z.B. für Berechtigungskonzepte und die Datenpflege. Zudem ist Entwicklung dieser SCL-Komponenten einfach zu erlernen und in Verbindung mit HTMLOUT auch schnell und einfach durchzuführen. In den SCL-Entries muß dann nur die Aufbereitung der Daten bzw. die Datenanalyse programmiert werden.



Für die Ausgabe wird grundsätzlich empfohlen, sich auf einfaches HTML 3.2 und GIFs für die Grafiken zu beschränken (zu Java siehe den vorhergehenden Abschnitt). Wenn die Ergebnisse vom Benutzer weiterverarbeitet werden sollen, bietet sich das Format "tab-separated-values" an, das z.B. direkt in ein Spreadsheet eingelesen werden kann. Solche Anwendungen sind vor allem dann sinnvoll, wenn mehrere der folgenden Kriterien zutreffen:

- Die mit HTML realisierbaren Menüsysteme bieten genügend Flexibilität, um alle Anforderungen der Benutzer darstellen zu können.
- Die Daten ändern sich so häufig oder den Benutzern sollen so viele Optionen angeboten werden, daß nicht alle möglichen Ergebnisse vorgerechnet werden können.
- Es sollen besondere Features, z.B. komplexere Berechtigungskonzepte, flexibel einzugebende Parameter etc., realisiert werden.
- Eine große Anzahl Benutzer, von denen aber jeder einzelne relativ selten auf die Informationen zugreift, so daß sich die Installation relativ teurer Software bei den Benutzern nicht lohnen würde.

- Es sollen "fremde" Benutzer angesprochen werden, oder bekannte Benutzer sollen von beliebigen Endgeräten auf die Daten zugreifen können (Internet, ggf. passwort-gesicherter Zugang).

3.1 Beispiel Korrelationskoeffizienten

In diesem Beispiel kann der Benutzer einen Dateinamen angeben und bekommt dann ein Menü angezeigt, wo er die Variablen für die Korrelationsmatrix auswählen kann. Nach der Anzeige der Korrelationsmatrix kann zu jedem Korrelationskoeffizienten ein Streudiagramm abgerufen werden.

3.2 Beispiel: Project Management System bei der EURESCOM

Ein Beispiel für den produktiven Einsatz der SAS/IntrNet Software mit HTMLOUT ist das Project Management System bei der EURESCOM GmbH, EMIS. Die EURESCOM GmbH ist eine Gründung von mittlerweile 24 Telecom-Gesellschaften in 23 Europäischen Ländern. Ihre Aufgabe ist es, Forschungsprojekte durchzuführen, mit dem Ziel, europaweit einheitliche modernste Telekommunikationsnetzwerke und -dienste aufzubauen und die Standardisierung der Technologien voranzutreiben. Mit der Intranet-Komponente von EMIS stehen alle Informationen, von Adresslisten über Budgets bis zu komplexen Auswertungen zur Budgetkontrolle an allen Arbeitsplätzen aktuell zur Verfügung. Dank geschützten Zugängen gilt das auch bei den nicht seltenen Auslandsreisen.

4. Zusammenfassung

Egal, ob Ergebnisse offline präsentiert oder auf Abruf online erstellt werden sollen, in den meisten Fällen stellt die WEB-Technologie mit HTML und GIF das geeignete Medium dar, um diese Aufgabe effizient und anwenderfreundlich zu lösen.

Mit der SAS-Software lassen sich die notwendigen Schritte von der Datenaufbereitung über die Analyse bis zur Ergebnispräsentation leicht durchführen und durch Macros oder AF-Programme unterstützen. Die SAS/IntrNet Software bietet mit dem Application Server eine stabile, robuste und gleichzeitig mächtige Basis für interaktive Informationssysteme im WEB, die sich nahtlos in statische Berichte integrieren lassen - und umgekehrt.

Mit HTMLOUT wird die Entwicklung dieser Anwendungen wesentlich erleichtert, in dem die Programmierung der Datenaufbereitung und Analyse von der Entwicklung der Präsentation in HTML getrennt werden. Dadurch wird eine relativ komplexe Aufgabe in relativ einfache und pflegeleichte Komponenten zerlegt.

Kontakt

Wilfried Schollenberger

WS Unternehmensberatung und Controlling-Systeme GmbH

Bergstraße 7

60120 Heidelberg

Tel: 06221 / 401 409

E-Mail: wisch@ibm.net

Web: www.t-online.de/home/ws-gmbh