

webScala - Ein System zur Datenanalyse und Ergebnisverwaltung

Alexander Jahnke
Siemens Healthcare Diagnostics
Emil-von-Behring-Str. 76
35041 Marburg
Alexander.jahnke@siemens.com

Zusammenfassung

webScala ermöglicht den Anwendern, Daten webbasiert zu analysieren und die Ergebnisse zentral abzulegen und zu verwalten. Über einen Webbrowser kann SAS Funktionalität als Stored Process im Intranet gezielt für bestimmte Benutzergruppen verfügbar gemacht werden, ebenso wie die damit erzeugten gespeicherten Reports.

webScala wurde von HMS Analytical Software auf Basis von SAS Business Intelligence Server als Ergänzung und langfristig als Nachfolger bereits langjährig etablierter SAS/AF-Anwendungen für Siemens Healthcare Diagnostics (ehemals Dade Behring Marburg GmbH) entwickelt.

Dabei kommen Lösungselemente auf Basis der folgenden Technologien zur Sprache:

- Ausführung von SAS-Programmen als Stored Processes
- Die Management Console aus dem SAS BI-Server
- Verwaltung der Berechtigungen bezüglich Anwendungen und Ergebnisse
- Ablage von Ergebnisse auf einem Webdav-Server

Schlüsselwörter: BI-Server, Webdav, SAS Management Console, webbasierte Analysen, Foundation Service, Stored Process, SCL

1 Zielsetzung

Im Bereich Pharma (In-Vitro Diagnostik) unterliegt die Datenanalyse in Forschung, Entwicklung und Produktion hohen Anforderungen an Datensicherheit, Nachvollziehbarkeit und Validierung. In Zeiten von Web2.0 gilt es auch im professionellen Umfeld, einem breiten Anwenderkreis die Funktionalität von SAS ohne Programmierkenntnisse zur Verfügung zu stellen. Dabei soll auf der einen Seite dem Programmierer die Möglichkeit gegeben werden, seine Lösungen schnell und ohne großen Aufwand bereit zu stellen und auf der anderen Seite der Anwender auf diese Lösungen ohne aufwendige Schulungen zugreifen können. Die Datenlage und -haltung ist bei Siemens Healthcare Diagnostics nicht dazu geeignet auf vorhandene Lösungen zu zugreifen. Während die klassischen SAS-Lösungen auf großen Datenmengen und einem zentralen Ablageort arbeiten, werden bei Siemens Healthcare Diagnostics viele kleine Datenmengen und dezentrale Ablageorte (bei jedem Anwender) genutzt. Als Datenquelle dienen i.d.R. Excel-Tabellen oder ASCII-Dateien. Mit webScala wird die Brücke zwischen BI-Techno-

logie, dezentraler Datenhaltung und der schnellen Bereitstellung von Lösungen durch innovative Webtechnologie geleistet.

2 Bisheriger Einsatz von SAS

2.1 SAS Analytics

Im Bereich der klinischen Studien und zulassungsrelevanten Daten für Laborsysteme wird SAS Analytics (Base, STAT, GRAPH) zur Auswertung und graphischen Darstellung der Ergebnisse genutzt. Dabei werden die Daten von Laboren und Studienzentren zur Verfügung gestellt und von erfahrenen SAS-Programmierern zur zentralen Projektbetreuung ausgewertet.

2.2 Auswertung mittels SAS/AF

Unter dem Namen SCALA (SAS Created Applications for Laboratory Analyses) sind eine Reihe von Modulen zur dezentralen Projektbetreuung zusammengefasst. Diese Module stellen mittels einer interaktiven Oberfläche in SAS/AF dem Anwender Auswertemethoden zur Verfügung. Unter anderem sind dies Methoden zur Varianzanalyse, Regressionsverfahren oder Interpolationsverfahren. Dabei können die Anwender auch ohne SAS-Kenntnisse die erforderlichen Funktionen und Parametrisierungen von SAS nutzen.

2.3 Archivierung der Berichte

Die als PDF erstellten Berichte werden auf einem Webserver in einer definierten Struktur abgelegt. Mittels eines SAS-Programms werden aus dieser Struktur statische HTML-Seiten geschrieben, die die Navigation innerhalb der Dokumente ermöglichen. Die Zugriffskontrolle wird über geschützte Verzeichnisse auf dem Webserver ermöglicht. Auf diesen Verzeichnissen werden benutzerabhängig Zugriffsrechte vergeben. Um die Verwaltung der Rechte überschaubar zu halten ist die Zugriffskontrolle auf drei Benutzerrollen pro Geschäftsbereich beschränkt. Es sind damit neun verschiedene Benutzerrollen möglich.

2.4 Bewertung

Die Applikationen im AF-Umfeld sind auf Dauer nicht zukunftssicher. Zum einen ist ein großes Know-how auf Seite der Entwickler nötig, zum anderen wird SAS/AF seit Jahren von SAS nicht mehr weiterentwickelt. Alternativen zur Oberflächenentwicklung in SAS sind praktisch nicht vorhanden. Der Bedarf für flexibel programmierbare Oberflächen zu SAS Funktionalitäten ist aber ungebrochen vorhanden.

Es gibt die Möglichkeit eine solche Oberfläche für SAS mit JAVA zu entwickeln, diese ist aber streng an eine (inzwischen veraltete) JAVA Version gebunden und kann in

Zeiten wöchentlicher Updates und Versionswechsel nicht produktiv eingesetzt werden. Zu dem fehlt die nachvollziehbare Verknüpfung zwischen Daten, Analyse, Parameter bis zum Ergebnis. Es kann nachträglich nur schwer feststellbar sein, mit welchen Daten und welchen Parametern ein bestimmtes Ergebnis erzeugt wurde.

Bzgl. der Archivierung von Berichten zeigt sich, dass die vergebenen Benutzerrollen zu unflexibel für den geforderten Bedarf einer projektorientierten Zugriffsweise sind. Da die Auswertefunktionalität und die Archivierungsfunktionalität getrennte System sind, ist die Administration des Archivierungssystems nur zentral und mit hohem Aufwand möglich.

2.5 Anforderungen an ein neues System

Aus diesen Vorbedingungen und Bewertungen ergeben sich die Anforderungen für eine neue Infrastruktur mit folgenden Randbedingungen:

- ✓ geringer **Installationsaufwand**
- ✓ geringer Aufwand zur Erstellung der **Benutzeroberfläche**
- ✓ kürzere **Bereitstellungszeiten** für Auswertefunktionalitäten
- ✓ **Vereinigung** von Auswertefunktionalität und Archivierungssystem
- ✓ **Nachvollziehbarkeit** der Auswertungen durch Hinterlegen der zugrunde liegenden Daten und Metadaten
- ✓ **Dezentrale** Vergabe der Dokumentattribute durch die Dokumentersteller
- ✓ **Spezifischere** Vergabe der Benutzerrollen statt auf Produktlinienebene nun auf Projektebene
- ✓ **Suchfunktion** über Keywords und Attribute wie z.B. Datum der Erstellung oder Zugriffsattribut
- ✓ **Audit Trail**

Die Liste der Anforderungen zeigt, dass es keine „Off-the-shelf“ Lösung von SAS gibt und dass eine dynamische Oberfläche mit dezentraler Datenhaltung nicht mit reinen SAS-Mitteln umgesetzt werden kann. Daher wurde webScala als ein maßgeschneidertes System von HMS Analytical Software GmbH in Heidelberg entwickelt.

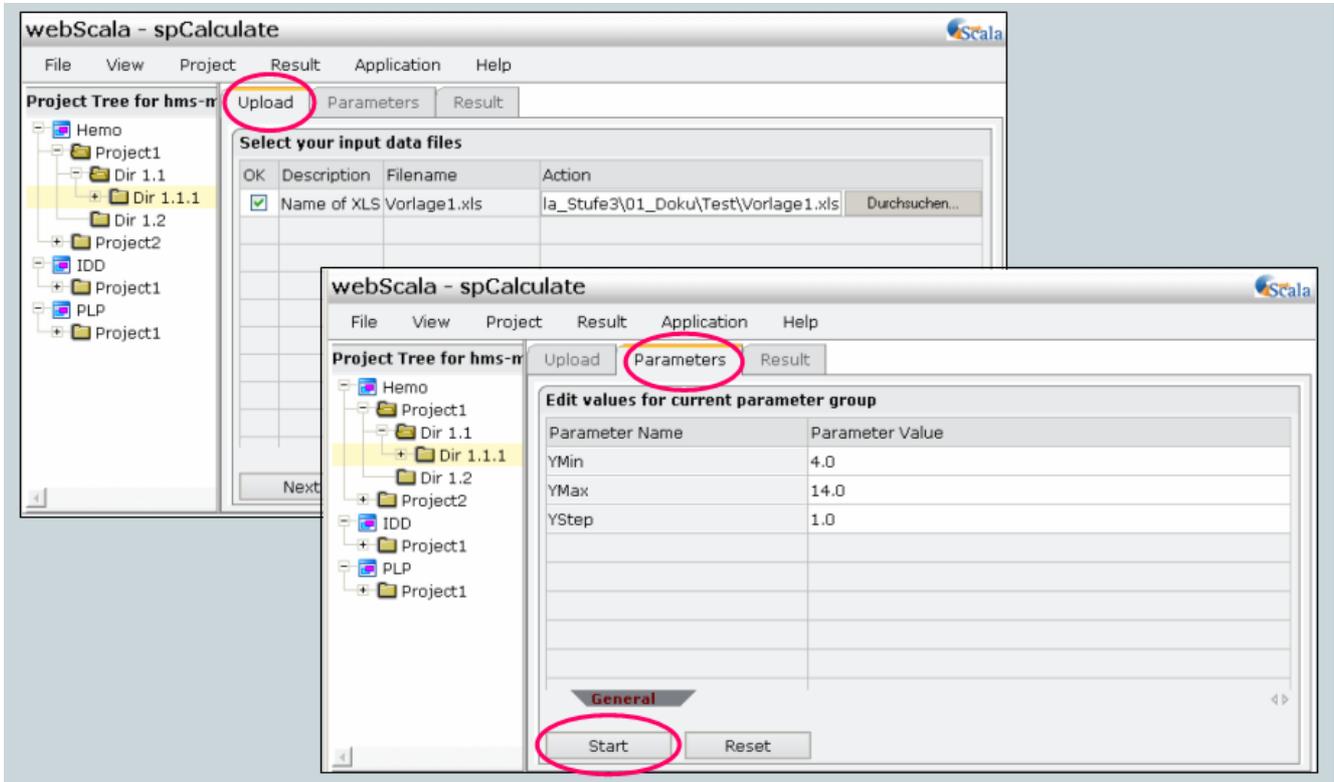
Die Oberfläche wurde mit dem Application Designer der Software AG erstellt und über die SAS Business API mit dem SAS BI Server verbunden. Eine Upload-Funktionalität ermöglicht es, dezentrale (Excel-)Dateien dem BI-Server zur Auswertung zur Verfügung zu stellen.

Dabei wurden folgende Spezifikationen gewählt:

- Webbrowser als Frontend
- SAS Programme werden als Stored Process (SP) bereitgestellt
- dynamisches Frontend zur Einbindung der Stored Processes
- Metadaten und Userverwaltung über BI-Server
- Speicherung aller relevanten Daten (Quelle, Ergebnis, Parameter)

- differenziertes Benutzerkonzept
- Logging der Meta-Daten für den Audit-Trail

3 webScala im Einsatz



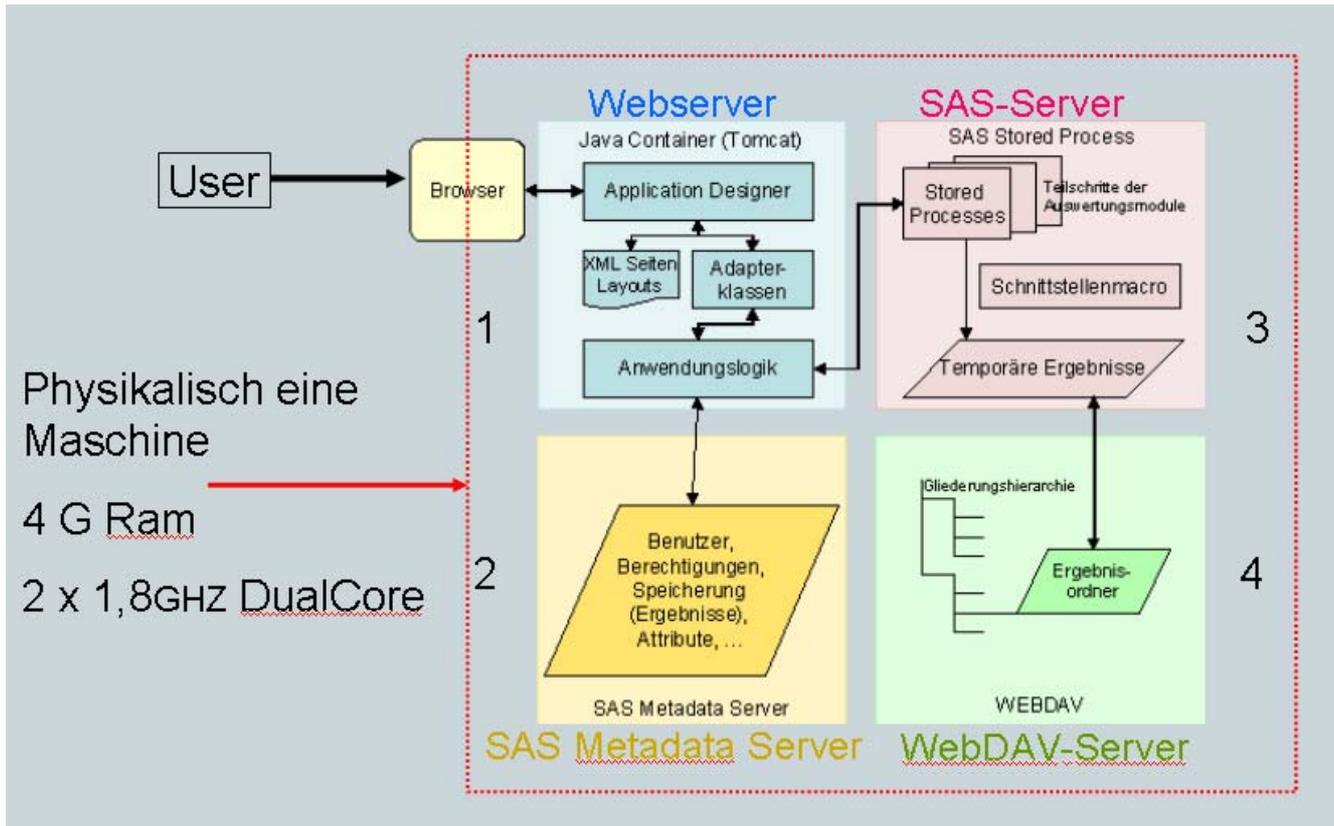
Beispiel-Screenshot mit Upload für Excel-Dateien und Parametrisierung eines Stored Prozesses. Ausführliche Screenshots befinden sich im Powerpoint Vortrag auf der Tagungs-CD

4 Architektur

Der User stellt per Browser eine Verbindung zu webScala her. Der Webserver (Apache) nimmt die Anfrage entgegen und leitet sie an einen Java-Container (Tomcat) weiter. Von dort aus wird die Authentifizierung gegen LDAP und die Autorisierung für webScala weiterverarbeitet.

Über den Application Designer wird die Anwendungslogik gesteuert und stößt von dort aus die Stored Prozesse auf dem SAS Stored Process Server an. Die so erzeugten Ergebnisse können per webDAV gespeichert werden.

Physikalisch laufen alle „Server“ (Webserver/Tomcat, MetaDaten-Server und Stored Process Server und webDAV Server) bisher auf einer Maschine mit 4 Gbyte RAM und 2 Dual-Core 1,8 Ghz Prozessoren.



4.1 Die Aufgaben der Komponenten im Einzelnen:

Stored Process:

Jedes SAS-Programm kann in einen Stored Process umgewandelt werden, dabei ist die Syntax ähnlich der eines SAS-Makros. Der Code wird zentral auf dem Server abgelegt und kann dort gepflegt werden.

SAS MetaDaten-Server

Stellt die Stored Prozesse bereit und verwaltet Benutzer, Berechtigungen, Ablageorte und Dateiattribute sowohl für die Stored Prozesse als auch die Ergebnisse.

WebDAV-Server

Ablage der physikalischen Dateien per Webanwendung, dabei werden die Ergebnisse der Auswertungen, ihre Parameter (in XML) und zusätzliche Dokumente gespeichert.

4.2 Kommunikation der einzelnen Komponenten

Als Schnittstelle zwischen den einzelnen Komponenten wird die SAS BI API (SAS Foundation Services) genutzt, die es ermöglicht mittels Java auf den SAS BI-Server zuzugreifen.

Die Spezifikation der SAS Foundation Services ist auf der Webseite von SAS zu finden:

<http://support.sas.com/rnd/gendoc/bi/api/index.html>

Die SAS Foundation Services sind eine Sammlung von Diensten für (Web-) Anwendungsentwickler, mit denen die Infrastruktur und Erweiterungen zur Verfügung gestellt werden, um beliebige Metadaten anzulegen, abzufragen oder Schnittstellen zu anderen Technologien aufzubauen (z. Bsp: Authentifizierung).

4.3 Services im Einsatz bei webScala

- Der Information Service wird genutzt, um Metadaten abzufragen und neue Daten anzulegen.
- Der User Service wird zur Authentifizierung genutzt und um Informationen über die Metadaten-Benutzer zu erhalten.
- Der Session Service verwaltet die Sitzungsdaten der einzelnen User und ordnet ihnen die richtigen Prozesse zu. Er wird auch benötigt um einen Stored Process auszuführen.

5 Endbetrachtung

Zum gegenwärtigen Zeitpunkt liegen noch keine Erfahrungen über den produktiven Einsatz von webScala vor. Daher können nur Erfahrungen aus der Prüfung und Entwicklung beschrieben werden.

Durch die am Windows Explorer orientierte Oberfläche zur Dokumentenablage bzw. – zugriff ist für den User eine intuitive Bedienbarkeit gegeben und er kann nach einer kurzen Einarbeitung den vollen Umfang der Applikation nutzen. Ebenso ist die dezentrale Administration durch (IT-)fachfremde User der einzelnen Projekte ohne großen Schulungsaufwand möglich. Neue Applikationen lassen sich ohne großen Aufwand in webScala einbinden und gezielt bestimmten Usergruppen zur Verfügung stellen.

Die von HMS entwickelte Oberfläche ist ungleich flexibler als die von SAS angebotenen Lösungen, so ist es zum Beispiel möglich dem Anwender dynamische, datenabhängige Auswahllisten zur SP-Parametrisierung anzubieten. Beispielhaft sei hier eine dynamische Auswahl aus einer hochgeladenen Excel-Datei genannt: webScala zeigt dem User alle enthaltenen Blätter aus dieser Datei und der User kann eines davon auswählen. Vom ausgesuchten Blatt werden nun die enthaltenen Spalten angezeigt und ermöglichen eine erneute Auswahl durch den User bis auf die Ebene der Unique Values.

Es ergaben sich allerdings auch einige kritische Punkte während der Prüfung, die ihre Ursachen in der Architektur unter SAS 9.1 haben:

- So ist eine Dokumentation der Parametrisierung beim Erstellen eines Stored Process nur aufwändig machbar, da es sich komplett um einen interaktiven Prozess mittels GUI handelt. Eine Parametrisierung per Script ist nicht möglich.
- Back-Ups sind nur als Tages-Roll-Back des gesamten Systems möglich. Gibt es also ein Problem in einem einzelnen Teilbereich bei nur einem User, müssen alle Daten (auch die nicht betroffenen Ergebnisse) auf den Tag vor dem Problem zurückgesetzt werden. Damit geht bei täglichem Back-Up, mindestens ein Tag an Ergebnissen verloren.
- Mit steigender Nutzung ist dem Thema Performance besondere Aufmerksamkeit zu widmen und die einzelnen Komponenten auf eigene dedizierte Server zu übertragen. In diesem Bereich fällt auch das Problem, dass ein einmal gestarteter SAS-Prozess auf dem Metadaten-Server nicht wieder gestoppt werden kann. Stossen also zu viele User einen rechenintensiven Prozess an und schließen danach die Anwendung ist der Server bis zu deren Abarbeitung beschäftigt.

Aktuell ist nicht absehbar, ob diese Probleme mit der Version 9.2 von SAS gelöst werden.