

15 WebFacing Programmiertipps

von Don Denoncourt

1. Konfigurationseinstellungen:

Einstellungen auf Anwendungsebene, wie beispielsweise IP-Adresse und Port des Host lassen sich in der Datei WEB/web.xml direkt auf der Einsatzplattform modifizieren.

2. Performance Tuning:

Zur Optimierung der Performance können die Einstellungen diverser web.xml Kontext-Parameter (wie z.B. die Bean Cache-Größe oder ein Abschalten der Komprimierung) verändert werden.

3. Detaillierung der Fehlermeldungen:

Der web.xml Kontext-Parameter WFEErrorJSPDetail erlaubt einen Wertebereich von 0 - 3. Je höher der eingetragene Wert ist, umso detaillierter sind die Fehlermeldungen in error.jsp.

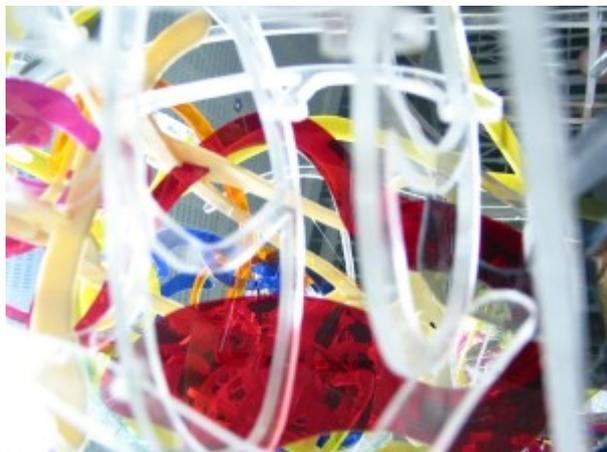
4. Server-Alternativen:

Da die WebFacing-Anwendung auf J2EE basiert, kann sie auf jedem Web-Anwendungs-Server auf beliebigen Plattformen ausgeführt werden. Für den Betrieb des TCP-WebFacing 5250-Server (STRTCPSVR *WEBFACING) sowie für die Ausführung der 5250-Anwendungen wird ein iSeries Server benötigt. Da der WebFacing-TCP-Server aber über einen TCP-Port (üblicherweise Port 4004) verfügbar gemacht wird, kann der Web- Anwendungs-Server auf jeder beliebigen Plattform laufen. Ich habe beispielsweise WebFacing-Anwendungen auf Apache Tomcat implementiert, ausgeführt auf einer Intel-basierten Linux-Plattform.

Artikelergänzung

Als Ergänzung zum Artikel „IBM WebFacing Tool - Ein Blick hinter die Kulissen“, erschienen in NEWSolutions, Ausgabe Mai 2005, lesen Sie hier 15 Programmiertipps.**Der Autor**

Don Denoncourt (denoncourt@comappsspec.com) ist WebSphere Consultant der Fa. Computer Applications Specialists, Inc. und technischer Autor für NEWSolutions. Übersetzt und für den deutschsprachigen Markt überarbeitet von Joachim Riener.



5. WebFace Link:

Der Code, der hinzugefügt werden muss, um einen Link zu der Website Ihres Unternehmens zu etablieren, ist aus der JavaScript-Funktion `launchApplication()` in `index.html` zu finden.

6. Anpassung der Logon-Seite:

Um eine angepasste Logon-Seite zu definieren, öffnet man die WebFacing-Standard Logon-Seite, wählt das Anzeigen des Quellencodes im Internet Explorer aus und sichert diese Seite in die Web Content Directory des WDS-Sc-Projektes als `logon.html`. Nach der Modifikation der Seite setzt man den `web.xml` Kontext-Parameter `WFUserDefinedSignon` auf „true“. Werden dabei JSP-Scriptlets eingesetzt, kann die modifizierte Seite optional den Namen `logon.jsp` erhalten.

7. Arbeiten mit dem WebFacing Job:

der spezifische `QQFINVOKER` Job eines Benutzers lässt sich mit der Anweisung `WRKUSRJOB` ermitteln.

8. Angepasste Server-seitige Verarbeitung:

Obwohl es möglich ist, den erstellten WebFacing Java Code nachträglich zu modifizieren, ist der Einsatz eines J2EE 1.3 Servlet-Filters sicherlich die bessere Strategie. Servlet-Filter erlauben die Aktivierung einer `web.xml` Einstellung, die dafür sorgt, dass Servlet Requests vor dem Aufruf des eigentlichen Servlet zuerst den definierten Servlet-Filter durchlaufen.

9. Anpassung der Funktionstasten JSP:

Zur Anpassung des Erscheinungsbildes der Funktionstasten wird der `web.xml` Kontext-Parameter `useCmdJSP` auf „true“ gesetzt. Anschließend werden die gewünschten Modifikationen in der Datei `CmdKeys.jsp` vorgenommen, die sich bereits in dem Verzeichnis `Web Content/styles/chrome` in dem WDS-Sc WebFacing-Projekt befindet.

10. Anpassung der Schaltflächen-Ereignisse:

Alle Funktionstasten-Schaltflächen sind für Aufruf von drei `webface.js` Routinen zur Abhandlung entsprechender Ereignisse vorbereitet. Nachdem diese IBM Funktionen ihre Verarbeitung abgeschlossen haben, können angepasste Routinen aufgerufen werden. Soll eine Schaltfläche eine besondere Funktion ausführen, müssen die folgenden Funktionen in einer angepassten JavaScript-Datei codiert werden:

Zugriff.

Noch nicht Abonnent? [Sonderaktion nutzen](#).

- [7 Euro/Monat NEWSabo digital - sofort zugreifen & online bezahlen.](#)
- [13,5 Euro/Monat NEWSabo plus inkl. 5x Logins & Print-Ausgaben - sofort zugreifen & per Firmen-Rechnung bezahlen.](#)