

Webframeworks - Teil 3

JSTL und Auswahl

AUTOR

Thomas Bayer
Orientation in Objects GmbH

Veröffentlicht am: 1.5.2003

ABSTRACT

Der dritte und letzte Teil dieses Artikels befaßt sich mit der Auswahl für ein Framework sowie der Beziehung zwischen Webframeworks und der **Java Standard Template Library** kurz JSTL.

) Schulung)

) Beratung)

) Entwicklung)

) Artikel)

Trivadis Germany GmbH

Weinheimer Str. 68
D-68309 Mannheim

Tel. +49 (0) 6 21 - 7 18 39 - 0
Fax +49 (0) 6 21 - 7 18 39 - 50

www.oio.dekontakt@trivadis.com

11 JSTL UND WEBFRAMEWORKS

Mit einfachen JavaServer Pages lässt sich bereits eine Trennung von Markup und Java Code erzielen. Code kann in *Helper Beans* und in *Taglibraries* ausgelagert werden. Eigene Tags, die sich in Server Pages verwenden lassen, können mit Java erstellt werden. Häufig benötigte Funktionalität für den Zugriff auf JavaBeans, XML und Datenbanken bietet die *JSP Standard Tag Library* kurz JSTL.

Die Funktionalität von JSTL überschneidet sich in einigen Punkten mit der von Templating Engines. Beispielsweise bietet die JSTL Tags für Schleifen und Bedingungen. Einige Web Frameworks stehen in Konkurrenz zur JSTL, andere werden von ihr bereichert. Das Struts Framework beinhaltet eine Reihe von Taglibs, die sich mit der JSTL überschneiden. Programmierer werden angehalten, die standardisierten Tags aus der JSTL anstatt der Struts spezifischen zu verwenden.

Die JSTL und eigene Taglibs sind kein Ersatz für ein Web Framework. Die Trennung von Markup und Code kann nicht allein durch Taglibs sichergestellt werden. Die JSTL oder eigene Taglibs können ein JSP basiertes Web Framework bereichern.

12 KAUFEN, BAUEN ODER OPEN SOURCE?

Neben der Verwendung von kommerziellen oder Open Source Frameworks gibt es die Alternative ein Framework selbst zu entwickeln. Bevor Struts ausgereift war, wurden einige MVC Frameworks individuell entwickelt. Heute würde ich die Entwicklung eines eigenen Web Frameworks nur bei sehr speziellen Anforderungen in Erwägung ziehen. Anders sieht es bei Frameworks aus, die auf bestehenden Frameworks basieren. Struts und Cocoon bieten die Möglichkeit der Erweiterung über Plugins oder spezielle *Haken* an denen man eigene Funktionalität einklinken kann. Auf diese Weise lässt sich ein spezielleres Framework *on Top of* entwickeln.

13 AUSWAHL EINES FRAMEWORKS

Die Auswahl für ein Web Framework ist eine strategische Entscheidung mit weitreichenden Auswirkungen auf ein Projekt oder Produkt. Alle Frameworks verwenden diverse Standards und Abstraktionen, die eine Migration zu einem anderen Frameworks erleichtern. Die Wahl für ein Framework führt zu einer dauerhaften Bindung, die nur unter hohen Aufwendungen gelöst werden kann. In die Entscheidung für ein passendes Framework sollte daher einige Zeit investiert werden.

Für Java gibt es eine Fülle von Web Frameworks. Legt man auf Stabilität und Verbreitung wert, so bleiben nur einige wenige Frameworks übrig. Ein Blick auf die Dokumentation und die Aktivität der Entwicklergemeinschaft, z.B. in Sourceforge genügt, um einige Frameworks als unbrauchbar zu entlarven.

Wichtig bei der Entscheidung für ein Framework, ist die Anzahl der bereits in Produktion befindlichen Installationen. Bei der Wahl eines Exoten kann es schnell passieren, dass Sie unbewußt als Pilotanwender dienen.

Ein wichtiges Kriterium für kommerzielle und freie Tools ist die Unterstützung von XML und J2EE Standards. Parser, XSLT Prozessoren, JDBC Treiber und weitere Komponenten sollten austauschbar sein. Die J2EE Muster sollten sich im Framework wiederfinden oder die Anwendung dieser Patterns unterstützen.

Eine pauschale Empfehlung für ein Framework gibt es nicht. Berücksichtigen Sie bei der Auswahl auch die Art der Aufgabe, die Teamgröße und den Kenntnisstand der Entwickler.

14 VORTEILE UND NACHTEILE

Für den Einsatz von Web Frameworks sprechen zahlreiche Vorteile:

- Höhere Produktivität
- Abschirmung von den Niederungen der Web Programmierung
- Auf fertige Dienste und Bausteine kann zurückgegriffen werden
- Bessere Aufgabenverteilung im Team
- Geführtes Programmiermodell
- Mehr Übersichtlichkeit bei großen Projekten
- Unterstützung durch Werkzeuge

Den Vorteilen stehen einige Nachteile gegenüber, die vor der Einführung eines Frameworks zu beachten sind:

- Das Angebot an Web Frameworks ist unüberschaubar
- Die Frameworks sind nicht leicht austauschbar
- Ein Framework erfordert Einarbeitung

15 FAZIT

Produktivität, Flexibilität und Arbeitsteilung im Team sprechen für sich

Oft lohnt sich bereits bei kleineren Web Anwendungen und Teams der Einsatz eines Frameworks. Mittlere und große Web Projekte können ohne Framework nicht wirtschaftlich realisiert werden. Die Vorteile Produktivität, Flexibilität und Arbeitsteilung im Team sprechen für sich. Web Frameworks sind eine wertvolle Hilfe für eine wirtschaftliche Entwicklung von Web Oberflächen.

Reife, Umfang und Dokumentation einiger Open Source Frameworks ermöglichen einen Einsatz in kommerziellen und kritischen Projekten. Nicht zuletzt sind beispielsweise Cocoon, Enhydra und Tapestry Vorreiter, die andere Produkte und Standards beeinflussen.

REFERENZEN

- Apache Cocoon
[\(http://xml.apache.org/cocoon/\)](http://xml.apache.org/cocoon/)
- Java/XML Application Server Enhydra
[\(http://www.enhydra.org/\)](http://www.enhydra.org/)
- Das Portalframework Jahia
[\(http://www.jahia.org/\)](http://www.jahia.org/)
- Jakarta Jetspeed: Framework für Enterprise Information Portale
[\(http://portals.apache.org/jetspeed-1/\)](http://portals.apache.org/jetspeed-1/)
- Jakarta Velocity
[\(http://velocity.apache.org/\)](http://velocity.apache.org/)
- Tapestry
[\(http://tapestry.apache.org/\)](http://tapestry.apache.org/)
- Turbine
[\(http://turbine.apache.org/\)](http://turbine.apache.org/)
- Espresso Application Development Framework
<http://www.jcorporate.com/html/products/espresso.html>
- WebMacro
<http://www.webmacro.org/> (<http://sourceforge.net/projects/webmacro>)