If we complicate things, they get less simple.
Professor an der Cambridge University

… Simplifications have had a much greater long-range scientific impact than individual feats of ingenuity. The opportunity for simplification is very encouraging, because in all examples that come to mind the simple and elegant systems tend to be easier and faster to design and get right, more efficient in execution, and much more reliable than the more contrived contraptions that have to be debugged into some degree of acceptability … Simplicity and elegance are unpopular because they require hard work and discipline to achieve and education to be appreciated.
Edsger W. Dijkstra

Complexity is a sign of technical immaturity. Simplicity of use is the real sign of a well designed product whether it is an ATM or a Patriot missile.
Daniel T. Ling

Simplicity is the soul of efficiency.
Austin Freeman, »The Eye of Osiris«

La perfection est atteinte non quand il ne reste rien à ajouter, mais quand il ne reste rien à enlever.
[Perfektion ist nicht dann erreicht, wenn nichts mehr hinzuzufügen ist, sondern wenn nichts mehr da ist, was weggenommen werden kann.]
Antoine de Saint-Exupéry

Controlling complexity is the essence of computer programming.
Brian Kernighan

Der Unterschied zwischen einem guten und einem schlechten Architekten besteht heute darin, daß dieser jeder Versuchung erliegt, während der rechte ihr standhält.
Ludwig Wittgenstein

Simplicity does not precede complexity, but follows it.
Alan J. Perlis

Vic Cekvenich · Wolfgang Gehner

Struts – Best Practices

Die Erstellung kommerzieller Web-Anwendungen
– mit Praxisbeispielen und Übungen

Übersetzt aus dem Amerikanischen
von Angelika Shafir

dpunkt.verlag

Vic Cekvenich
info@basebeans.com

Wolfgang Gehner
info@infonoia.com

Übersetzung: Angelika Shafir, Tel Aviv, Israel
Lektorat: Michael Barabas
Copy-Editing: Ursula Zimpfer, Herrenberg
Satz und Herstellung: Verlagsservice Hegele, Dossenheim
Umschlaggestaltung: Helmut Kraus, Düsseldorf
Druck und Bindung: Koninklijke Wöhrmann B.V., Zutphen, Niederlande

Bibliografische Information Der Deutschen Bibliothek

Die Deutsche Bibliothek verzeichnet diese Publikation
in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten
sind im Internet über <http://dnb.ddb.de> abrufbar.

ISBN 3-89864-284-4

1. Auflage 2004

Ringstraße 19 b
69115 Heidelberg

5 4 3 2 1 0

Vorwort

Sind Sie Mechaniker oder Fahrer?

Manche Leute wachen frühmorgens auf und gehen in ihre Garage. Dann legen sie sich unter das Auto, um sich den Motor vorzuknöpfen. Sie interessieren sich für den Unterschied zwischen Viertaktern und Kreiskolbenmotoren, für Ventilsteuerung, Einspritzpumpe, Arbeitstakt und ähnliche Dinge.

Andere Leute wachen am Morgen ebenfalls auf und gehen in ihre Garage. Sie fahren mit dem Auto zur Arbeit und starren auf die Autobahn. Sie machen sich normalerweise keine Gedanken darüber, in welchem Takt sich der Motor im Augenblick dreht, was der Auspuff macht oder wie's mit dem Einspritzsystem aussieht.

In der Programmierung gibt es ebenfalls zwei Gruppen von Menschen: Systemprogrammierer und Anwendungsprogrammierer.

Systemprogrammierer schreiben Code, der von Anwendungsprogrammierern benutzt und ausgewertet wird; sie schreiben Frameworks oder DAO-Implementierungen. Applikationsprogrammierer schreiben dagegen für Endbenutzer – um *Geschäftswert zu liefern*. Sie nutzen die Arbeit anderer Leute, so dass sie sich auf die Applikation konzentrieren können und sich nicht um die Technik (Mechanik) kümmern müssen.

Applikationen lassen sich effizienter schreiben, wenn man ein Framework und andere wiederverwendbare JARs sowie Komponenten ausnützt, statt das Rad jedes Mal neu zu erfinden.

In diesem Buch lernen Sie, wie Sie mit Hilfe des Open-Source-Frameworks Jakarta-Struts

- *qualitativ hochwertige* Web-Applikationen
- *sehr viel schneller* und
- *kosteneffektiv*

konstruieren können.

Das Jakarta-Struts-Framework ist ein Projekt der Apache Software Foundation (`www.apache.org`). Es bietet eine stabile Applikationsarchitektur und reduziert enorm die Zeit, die erforderlich ist, um Web-Applikationen zu schreiben, die Wertschöpfung und damit Geschäftswert liefern.

Struts basiert auf veröffentlichten Standards und bewährten Design-Paradigmen, wie z.B. dem Model-2-Ansatz, einer Variante der etablierten MVC-Archi-

tektur (Model-View-Controller; wird seit den siebziger Jahren eingesetzt), die mit der *J2EE*-Plattform (*Java 2 Enterprise Edition*) von Sun kompatibel ist.

Zusammen mit der *Java Standard Tag Library* (*JSTL*) bringt Struts *JavaServlets*, *JavaBeans*, *Java Server Pages* (*JSP*), *Extensible Markup Language* (*XML*) und Message-Ressourcen in ein einheitliches Framework ein.

Was Sie in diesem Buch lernen

Im ersten Teil des Buchs (Grundlagen) lernen Sie, den Weg für die Entwicklungsphase mit Struts und JSTL zu pflastern. Dieser Teil behandelt Software- und Hardwareanforderungen, solide Anforderungsspezifikationen, Datenbankeinrichtung, die Verwendung einfacher Actions, Beans und JSP und gibt Tipps zur Projektplanung.

Im zweiten Teil (Bausteine) erfahren Sie, wie man eine solide und leistungsstarke Applikationsinfrastruktur konstruiert und die Vorteile umfangreicher Wiederverwendung ausschöpft. Mit dieser Infrastruktur erhalten Sie ein Mittel an die Hand, um solide Applikationsmodule sehr schnell und effizient zu erstellen.

Im dritten Teil (Perspektiven) lernen Sie, wie man eine umfangreiche Benutzeroberfläche bereitstellt, die erweiterte Sicherheit bietet, und wie man eine Applikation zukunftssicher macht und dafür sorgt, dass sie in einer Produktionsumgebung gut skaliert.

Best Practices

Jeder kann eine Brücke bzw. eine Web-Applikation konstruieren. Ein Ingenieur kann Web-Applikationen aber erst durch Kenntnis (und Anwendung!) von Best Practices optimal auslegen, d.h. beste Qualität, kürzeste Zeit, geringste Kosten.

Die Autoren haben das klägliche Misslingen von Entwicklungsprojekten gesehen, wenn Architekten und Entwickler die hervorragenden J2EE-Blueprints fälschlicherweise als einzige Orientierung herangezogen haben. Deshalb deckt dieses Buch nicht alles ab, was man mit J2EE und Struts machen kann. Vielmehr konzentriert es sich auf die Best Practices, bei denen es sich gezeigt hat, dass sie sich in der realen Welt am besten eignen und letztendlich zu solchen Web-Applikationen führen, die den höchsten Geschäftswert in Bezug auf Qualität, Geschwindigkeit und Kosten bieten.

Dieses Buch greift beispielsweise auf, wie man die Entwicklungsproduktivität durch Maximierung der Wiederverwendung von Beans, Actions und Ereignissen steigert, wie man die Zuverlässigkeit einer Applikation durch geschickte Auswahl von Software und Hardware erhöht und wie wichtig eine solide Anforderungsspezifikation und Projektplanung für die kosteneffektive Entwicklung und Erfüllung der Kundenanforderungen sind.

»Fools ignore complexity, experts avoid it; geniuses remove it.« (Alan Perlis)

Die Autoren behaupten nicht, Genies zu sein. Sie haben lediglich versucht, das Buch einfach und kurz zu halten, indem sie sich auf die Ansätze konzentrierten, die in der Praxis am besten funktionieren und den gesamten Projektzyklus abdecken.

Nachfolgend sind die 11 Best Practices aufgeführt, die schnell und kosteneffektiv zur Produktion von qualitativ hochwertigen Web-Applikationen beitragen. Sie werden in den nächsten Kapiteln behandelt:

1. Erfülle die Kundenanforderungen effizient durch solide Anforderungsspezifikationen.
2. Spare Geld und erhöhe die Zuverlässigkeit mit effektiver Open-Source-Software.
3. Arbeite iterativ mit einem bewährten Prozess an der Entwicklung.
4. Führe Unit Tests durch.
5. Verwende MVC (Model-View-Controller).
6. Nutze objektorientierte Programmierung, um Wiederverwendung zu maximieren.
7. Verwende DAO (Data Access Objects).
8. Verwende CRUD-Ereignisse (Create-Read-Update-Delete).
9. Verwende CMA (Container Managed Authorization).
10. Führe Belastungstests durch.
11. Verwende Qualitätssicherung (QA) und verwalte Releases.

Praxisbezogenheit

Mit diesem Buch sollen Sie sofort loslegen können, es stützt sich also stark auf Lernübungen. Um wirklich einen Nutzen aus diesem Buch zu ziehen, sollten Sie die Übungen nicht überspringen. Sie helfen Ihnen, die Materie besser zu verstehen und Vertrauen in die demonstrierten Technologien zu gewinnen[1].

Wenn Sie mit dem Buch und den Übungen fertig sind, können Sie auf Wunsch Ihre neu erworbenen Fähigkeiten mit basicPortal CMS erproben. Es wurde mit Hilfe der in diesem Buch behandelten Best Practices erstellt.

basicPortal CMS ist eine dynamische, Struts-basierte Portalanwendung mit Content Management, die die Funktionalitäten und Features auf sich vereint, die in etwa 80% aller Web-Projekte benötigt werden. Dadurch können Sie sich auf die Features konzentrieren, die speziell in Ihrem Web-Projekt vorkommen.

basicPortal unterstützt E-Commerce bzw. Kreditkarten, Nachrichten, Lead-Tracking, Content-Syndication, Foren, Kalender, Web-Logs (»Blogs«), Wikis, E-Mail, schnelle standardisierte J2EE-Sicherheit, zeilenbasierte Sicherheit, Bilder, Blobs und Uploads.

1. Der Begleit-Code für die Übungen ist zum Download erhältlich. Näheres hierzu siehe Seite 4 »Projekt einrichten«.

Warum Struts?

Um die Entwicklung zu beschleunigen, können Sie ein proprietäres Framework erwerben. Möglicherweise stellen Sie aber fest, dass der Anbieter es Ihnen nicht gestattet, die Motorhaube zu öffnen, um das Öl zu wechseln. Jakarta-Struts ist Open Source und lässt Sie unter die Haube gucken. *Sie* haben die Kontrolle über den Quellcode und können ihn ändern, um eine Web-Applikation zu konstruieren, die die Anforderungen Ihres Kunden voll erfüllt.

Abgesehen davon bietet Struts weitere Vorteile, sowohl für Entwickler und Programmierer als auch für das Geschäft Ihres Kunden.

Hier die wichtigsten Vorteile für den Entwickler:

- Struts läuft auf jedem J2EE-Applikationsserver, darunter *IBM WebSphere*, *BEA WebLogic*, *Oracle IAS*, *Borland*, *Novell exteNd*, *Tomcat*, *Resin* und *Orion*.
- Struts ist eine extrem solide und stabile Architektur (wobei diese Stabilität durch Codebeiträge der Gemeinde noch gesteigert wird).
- Struts lässt sich leicht anwenden, ist aber dennoch für große, datengesteuerte Web-Applikationen geeignet.
- Struts zerlegt komplexe Applikationen in einfache, konsistente Komponentenmengen.
- Struts stellt sicher, dass alle Entwickler eines Teams nach dem gleichen Ansatz programmieren.
- Zu Struts gibt es eine komplette Dokumentation (Benutzer- und Entwicklerhandbücher).
- Struts beruht auf Best Practices, weil viele Benutzer am Debugging mithelfen und den Open-Source-Code verbessern.
- Struts besitzt eine große Entwicklergemeinde, die praktisch sofortigen Support leistet.

Darüber hinaus bietet Struts folgende Wertschöpfungsvorteile:

- Schnelle und kosteneffektive Entwicklung
- Wesentlich schnellere Marktgängigkeit und Produkt-Releases
- Open Source: Die Lizenz ist entweder kostenlos oder zu einem symbolischen Betrag zu erwerben.
- Unabhängigkeit von geschlossenen, proprietären Systemen und folglich keine teuren Lizenzen, die verwaltet, erneuert und aufgerüstet werden müssen.
- Keine Abhängigkeit von Spezialistenarbeit und kommerziellen Anbietern
- Zahlreiche Features erfüllen eine Vielzahl von Geschäftsbedürfnissen.
- Plattformunabhängig
- Höchst modular

Warum Sie dieses Buch lesen sollten

Dieses Buch wurde für erfahrene Java-Entwickler und -Programmierer geschrieben, die entweder in einem bereichsübergreifenden IT-Team oder für einen internen Kunden oder auch als unabhängige Berater für externe Kunden arbeiten.

Wenn Sie als unabhängiger Berater für das Kundenprojekt von A (wie Analyse der Benutzerbedürfnisse) bis Z (wie Zusammenstellung des vollständigen Projekts) zuständig sind, dann sollten Sie alle Projektphasen verstehen und selbst dazu in der Lage sein, sie zu implementieren.

Wenn Sie zu einem internen bereichsübergreifenden IT-Team gehören, kann es ratsam sein, auch die Aufgaben und Zuständigkeiten der anderen Teammitglieder, wie z.B. Projektleiter, Systemanalytiker, Architekt und technische Support-Mitarbeiter, zu verstehen.

Wenn ein Projekt fehlschlägt, werden allzu oft die Entwickler oder Programmierer beschuldigt. Im langfristigen Interesse der ganzen Zunft und in Ihrem eigenen Interesse kann es deshalb nicht schaden, kompetent zu sein und bessere Ansätze vorschlagen zu können, z.B. auf einer vollständigen Anforderungsspezifikation vor der Entwicklung zu bestehen, eine Softwareauswahl zu empfehlen oder sich für eine effektive Projektmanagementtechnik auszusprechen.

Dieses Buch eignet sich hervorragend für Teamarbeit. Seien Sie offen dafür, Informationen und Erfahrungen mit Ihren Kollegen auszutauschen. Falls Sie derzeit oder künftig an einem großen Projekt beteiligt sind, kann das Konstruieren und Arbeiten an der Grundlage eines gemeinsamen Verständnisses von Technologien für den Erfolg des Projekts Wunder wirken.

Was der Leser wissen sollte (Voraussetzungen)

Um aus diesem Buch den größten Nutzen zu ziehen, sollte Ihnen das MVC-Paradigma (Model-View-Controller) bekannt sein. Abbildung 1 zeigt die MVC2-Architektur, die wir mit Struts und JSP verwenden werden.

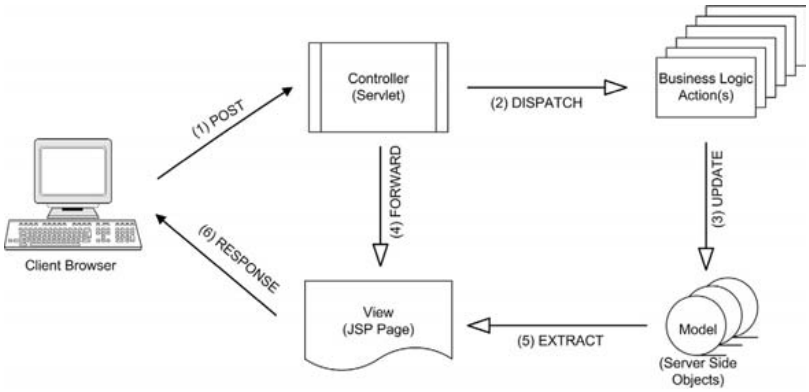

Abbildung 1: *Das MVC-Paradigma*

Sie sollten den Request-Response-Zyklus und die Terminologie kennen. Der Ablauf ist wie folgt:

1. Der Client-Browser postet eine Anfrage (Request) an das Controller-Servlet.
2. Das Controller-Servlet schickt sie an eine Action-Klasse.
3. Die Action bringt das Modell an die richtige Stelle (d.h., es liest Daten, aktualisiert sie, fügt Daten ein oder löscht sie).
4. Der Controller leitet das Ganze an eine JSP-Seite weiter.
5. JSP extrahiert die positionierten Daten aus dem Modell.
6. Der Controller gibt den JSP-Inhalt als Antwort (Response) zurück.

Sie sollten wissen, was JSPs sind und wie sie funktionieren.

Sie sollten auch einige Erfahrung mit dem Java-Servlet-API sowie mit HTML, SQL und JavaScript haben.

Je mehr Java-Programmiererfahrung Sie mitbringen, umso besser. Im Idealfall haben Sie bereits an der Entwicklung einer Java-Applikation, die es in die Produktion geschafft hat, mitgearbeitet.

Schnelltest 1: Navigieren Sie im Internet zu `http://java.sun.com/products/servlet/2.3/javadoc/index.html`.

Sind Sie mit dem API vertraut?

Schnelltest 2: Verstehen Sie das folgende HTML-Fragment?

```
<FORM ACTION=http://localhost:80/myServlet>
First Name: <INPUT TYPE="TEXT" NAME="firstName"><BR>
Last Name: <INPUT TYPE="TEXT" NAME="lastName"> <BR>
<INPUT TYPE="SUBMIT">
</FORM>
```

Wenn die Antwort auf beide Fragen »nein« lautet, empfehlen wir Ihnen, sich zuerst in die relevanten Themen einzulesen oder ein Teammitglied zu finden, das Ihnen die erforderlichen Grundlagen beibringt.

Im Buch benutzter Code und Software

In diesem Buch verwenden wir Standardprogrammiertechniken, die mit jeder gewählten Servlet-Engine, jedem Applikationsserver, jeder SQL-Datenbank und jedem IDE funktionieren.

Dieses Buch ist eine Art Kochbuch, weil viele Projekte nicht nur Struts, sondern auch andere Softwarezutaten nutzen. Wann immer andere »Zutaten« notwendig sind, wird in diesem Buch einer Open-Source-Software gegenüber proprietärer Software der Vorzug gegeben, weil Open Source zahlreiche Vorteile bietet, wie wir im weiteren Verlauf noch sehen werden.

Keine einzelne Softwarefirma, egal, wie groß sie sein mag, kann alle Werkzeuge in der jeweils besten Form selbst entwickeln und vertreiben. In der Open-Source-Gemeinde gibt es einfach zu viele »Köpfe«, um mit deren Errungenschaften ständig gleichziehen oder sie gar übertreffen zu können.

Geistiges Eigentum und Lizenz

Dieses Buch diskutiert und demonstriert die Verwendung lizenzierter Produkte. Die meisten sind Open-Source-Lizenzen. Die Softwareprodukte in diesem Buch (insbesondere Jasic und dessen Code) werden nur zur Wissensvermittlung diskutiert und zum Download zur Verfügung gestellt. Mit der Demonstration eines Beispiels erwerben Sie also nicht das Eigentum daran. Sie müssen sich informieren, wem sie gehören und wofür Sie lizenziert sind.

Wichtiger Hinweis: Alle Entwürfe und Softwareprodukte unterliegen dem jeweiligen Copyright; Entwürfe sind patentiert, insbesondere das basicPortal-Datenmodell; seine Applikationen und Basisklassen sind patentiert, jedoch im Rahmen einer flexiblen Lizenz erhältlich.

Über die Autoren

Vic Cekvenich verfügt über 14 Jahre Erfahrung in führenden IT- Positionen und ist auf Projekt-Recovery spezialisiert. Er besitzt mehr als zehn Zertifizierungen in Bereichen wie Java, SQL, objektorientierte Programmierung und Entity-Relationship-Modeling. Er hat ein Buch über die Verwendung des MVC/Struts-Frameworks mit Data Access Objects (DAO) veröffentlicht und viele Arbeiten und Artikel über weitere Themen in diesem Bereich geschrieben. Er wurde vom Java Developer's Journal als »Trainer of the Year« ausgezeichnet und war auf der Comdex, NetWorld, Windows World und vielen anderen wichtigen Fachmessen präsent.

Vic Cekvenich hat für die NASA, CSC, Ford Motor Company, Bank of America, Perot Systems, BP-Amoco und andere Unternehmen in verschiedenen Positionen gearbeitet, die von der technischen Projektleitung bis zum Vice-President von Managementinformationssystemen (MIS) reichen. Zu seinen Erfahrungen zählen Arbeiten mit großen Teams, Etats von über einer Million Dollar und Datenbanken in Terabyte-Größe mit über 40.000 Benutzern.

Vic Cekvenich ist einer der maßgeblichen Entwickler bei baseBeans Engineering. Sein derzeitiges Projekt, das basicPortal-System, ist eine datenbankgesteuerte Web-Applikation mit Fokus auf Leistung und Bedienungsfreundlichkeit.

baseBeans Engineering (`www.baseBeans.com`) bietet kommerzielle Unterstützung für basicPortal-Benutzer. Entwickler sind eingeladen, ihre eigene Unterstützung einfließen zu lassen und jederzeit individuelle Module zu entwickeln.

Vic Cekvenich erhielt den JDJ Readers' Award für das »Best Java Training Program«; er unterrichtet bereits seit über neun Jahren im Entwicklungsbereich.

Wolfgang Gehner verfügt über 13 Jahre Erfahrung im Projektmanagement, u.a. von Multi-Millionen-Euro-Projekten, und hat 6 Jahre Erfahrung als Softwareentwickler, Analyst, Architekt und Trainer bzw. Coach.

Wolfgang Gehner ist Gründer und Chief Executive Officer der Firma Infonoia SA (Information ohne Paranoia) mit Sitz in Genf in der Schweiz.

Infonoia (`www.infonoia.com`) entwickelt eine serverbasierte Enterprise Document Management Suite. Sie umfasst Web-Applikationen für intelligentes Dokumentenmanagement, Business Intelligence, Business Process Management und Content Management sowie Dokumentenbanken auf intelligenten, servergesteuerten CD-ROMs und DVDs.

Der technologische Schwerpunkt liegt dabei auf vom Anwender unterstützten Lernalgorithmen, die für intelligente Dokumentenanalyse, inkrementielles Web Crawling und -Monitoring sowie Textindizierung eingesetzt werden.

Die Firma Infonoia war erster Kunde und kommerzieller Anwender von Applikationsservertechnologien in der Schweiz und einer der ersten in Europa. Seit 2001 wendet Infonoia Struts aktiv in der Entwicklung von Web-Applikationen an.

Als Chefarchitekt von Infonoia zeichnet Wolfgang Gehner für Kunden verantwortlich wie Procter & Gamble, Deloitte & Touche, LVMH, Tetra Pak, die Weltgesundheitsorganisation, die Universität Bielefeld, die Republik und den Kanton Genf und Lombard Odier Darier Hentsch.

Kontaktinformationen

Wenn Sie Fragen oder Kommentare haben, kontaktieren Sie bitte die Autoren:

- Vic Cekvenich (`info@basebeans.com`) oder `www.baseBeans.com`.
- Wolfgang Gehner (`info@infonoia.com`) oder `www.infonoia.com`.

Danksagung

Unser besonderer Dank gilt unseren Lektoren Boris Baginski, Henning Brune, C. DeFazio und Taylor Cowan.

Inhaltsverzeichnis

Teil II – Bausteine