Zeitschrift: bulletin.ch / Electrosuisse
Herausgeber: Electrosuisse
Band: 99 (2008)
Heft: 5

Artikel: Engineering Doodle
Autor: Sevinç, Paul E. / Nöf, Michael
DOI: https://doi.org/10.5169/seals-855828

Engineering Doodle

Wie heute ein Webdienst programmiert wird

Über Doodle lassen sich Termine finden für Geschäftssitzungen, Abendessen mit Freunden oder diverse andere Gelegenheiten. Es ist ein kostenloser Webdienst, der sich grösster Beliebtheit erfreut – und notabene aus der Schweiz kommt. Zwei seiner Markenzeichen sind seine Einfachheit und die Tatsache, dass seine Nutzung keine Benutzerregistrierung voraussetzt. Was nach aussen kaum komplexer wirkt als ein Beispiel aus einem Lehrbuch zur Webprogrammierung, wird im Hintergrund modernsten Ansprüchen des Security- und Software-Engineerings gerecht.

Paul E. Sevinç, Michael Näf

Am Anfang stand der Termin – oder eben nicht. Nach mehreren erfolglosen Versuchen, per E-Mail einen Termin zu finden, um mit Kollegen abzumachen, gab Michael Näf entnervt auf. Daraufhin schlug einer der Kollegen vor, eine Excel-Datei von einem Kollegen zum nächsten zu schicken, damit diese darauf ihre Verfügbarkeit ankreuzen könnten. Michael fand die Idee bestechend, aber auch verbesserungswürdig. So nahm er sich ein Wochenende lang Zeit, ein Datenbankschema zu entwerfen und ein Perl-Skript zu implementieren, um Datumstabellen webbasiert zur Verfügung zu stellen. Doodle war geboren (Bild 1).

Mit Modifikationen und Ergänzungen leistete das Originalsystem von 2003 bis im Sommer 2007 zuverlässig seinen Dienst. Abgelöst wurde es durch ein in Java implementiertes System. Zuvor hatte Michael mit Paul E. Sevinç die im Technopark Zürich ansässige Inturico Engineering GmbH gegründet, um Doodle zu professionalisieren.

Architektur

Doodle ist eine typische dreistufige Anwendung (three-tier application): Benutzer kommunizieren über ihren Browser mit dem Webserver, der wiederum mit dem Datenbankserver kommuniziert (Bild 2). Bis Dezember 2007 war die Trennung zwischen Webserver und Datenbankserver rein konzeptioneller Natur. Seit Januar 2008 ist sie auch physischer Natur, um Doodles Wachstum auch in Zukunft gerecht zu werden.

Bei den Browsern wird vorausgesetzt, dass sie CSS und XHTML interpretieren können, womit aktuelle Versionen von Firefox, Internet Explorer oder Safari keine Mühe haben. Dass sie Cookies und JavaScript (geschweige denn Flash oder andere Plug-ins) unterstützen, wird nicht vorausgesetzt. Damit kann Doodle auch dort eingesetzt werden, wo aus Gründen des Datenschutzes oder der Sicherheit auf Cookies und JavaScript verzichtet wird, wie es in vielen Unternehmen der Fall ist.

Als Datenbankverwaltungssystem (database management system, DBMS) fungiert zur Laufzeit MySQL. Während der Entwicklung und dem Testen kommen auch HSQLDB und die H2 Database Engine zum Einsatz. Weshalb Doodles mittlere Stufe (middle tier) auf verschiedenen Datenbankverwaltungssystemen aufsetzen respektive deren SQL-Dialekt sprechen kann, wird gleich nachfolgend erläutert.

Model

Doodles Kern ist die mittlere Stufe, die im Wesentlichen aus drei Schichten (layers) besteht (Bild 3). Die unterste Schicht ist die Model-Schicht mit Schnittstellen für Umfragen, Teilnehmende etc. Davon gibt es eine konkrete, auf Hibernate basierende Implementation. Alternative Modellimplementationen sind denkbar, beispielsweise auf Basis von Amazons SimpleDB-Dienst.

Hibernate ist ein von JBoss, einer Division von Red Hat, gefördertes Open-Source-Projekt. Hibernate abstrahiert einerseits das konkrete DBMS (in Doodles Fall wie oben erwähnt MySQL, HSQLDB und H2 Database Engine). Andererseits vermittelt Hibernate in beide Richtungen zwischen Javas Objektmodell und dem relationalen Modell des DBMS. Hibernate ist äusserst mächtig, aber entsprechend auch komplex. Wenigstens wird einem der Einstieg durch zahlreiche Hibernate-Publikationen leicht gemacht. Trotz steiler Lernkurve hat sich der Einsatz von Hibernate für Doodle schnell gelohnt. Heute würde sich höchstens die Frage stellen, direkt auf die Java Persistence API (JPA) zu setzen und Hibernate nur als eine von mehreren möglichen JPA-Implementationen zu nutzen.

View

Die oberste Schicht ist die View-Schicht. Das von Browsern zu interpretierende XHTML wird dynamisch generiert mit Java-Server Faces (JSF) in Kombination mit Facelets. Aber Servlets kommen auch direkt zum Einsatz, beispielsweise um auf Basis des Abdera-Projekts der Apache Foundation Atom-Feeds zu generieren, worauf wir in diesem Artikel jedoch nicht eingehen.

JSF ist eine von Sun Microsystems geförderte Spezifikation für webbasierte Benutzerschnittstellen, die auf Ereignisse reagieren, wie man es sich als Entwickler von grafischen Benutzerschnittstellen (graphical user interfaces, GUIs) gewohnt ist. Die Objekte, welche die Ereignisse verarbeiten, sind zwar «plain old Java objects» (POJOs), aber einen kleinen Preis muss man für die Ereignissteuerung doch bezahlen: Sämtliche JSF-Anwendungen hängen von einer Sitzung (session) ab, selbst solche, die für

	April 2008		Mai 2008		
	Mo 28	Mi 30	Mo 5	Mi 7	Fr 9
	08:00	10:00	08:00	10:00	15:00
Guido Santner	OK	OK	OK	OK	OK
Rolf Schmitz			OK	OK	
Paul Sevinç	OK	OK	OK	OK	OK
Michael Näf	OK		OK	OK	OK
Ruedi Felder				OK	OK
Jörg Weber		OK	OK		OK
Ihr Name	☐	☐	☐	☐	☐
Anzahl	3	3	5	5	5

Bild 1 Mit Doodle Sitzungstermine abmachen.

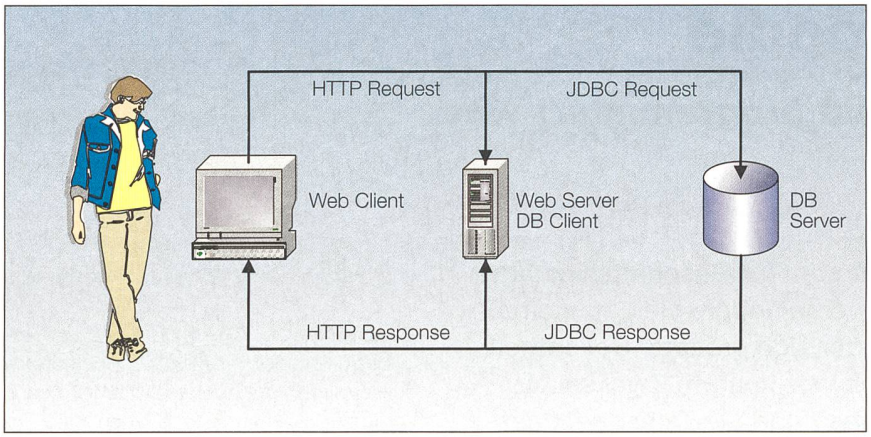


Bild 2 Architektur von Doodle.

die Geschäftslogik keiner Sitzung bedürften. Im Fall von Doodle hat sich das an einigen wenigen Stellen unschön manifestiert, weshalb wir dort JSF umgehen, was sicher nicht im Sinne des Erfinders war.

Von den erhältlichen JSF-Implementationen dürften die Referenzimplementation von Sun Microsystems und MyFaces von der Apache Foundation die verbreitetsten sein. Für Doodle setzen wir MyFaces ein, ursprünglich in der Version 1.1 gemäss der JSF-Spezifikation 1.1 und mittlerweile in der Version 1.2 gemäss der aktuellen JSF-Spezifikation 1.2. Die Aktualisierung war innert einer Stunde gemacht.

Ein Manko von JSF sind Schablonen (templates). Dies ist allerdings nicht als Vorwurf zu verstehen; tatsächlich bietet JSF mannigfaltige Möglichkeiten der Erweiterung. Entsprechend können diverse Schablonenlösungen in Kombination mit JSF eingesetzt werden, beispielsweise das aus dem Struts-Projekt herausgelöste Tiles der Apache Foundation. Wie bereits erwähnt, setzen wir für Doodle Facelets ein. Facelets ist ein Open-Source-Projekt, das – wie es sein Name andeutet – im Hinblick auf JSF lanciert wurde und unsere sämtlichen Anforderungen mit Bravour erfüllt. Leider können wir uns des Verdachts nicht erwehren, dass die Weiterentwicklung von Facelets etwas stockt. Umso interessierter beobachten wir, wie sich JSFTemplating entwickelt, das ebenfalls ein im Hinblick auf JSF lanciertes Open-Source-Projekt ist.

Wer sich in letzter Zeit mit Webapplikationen im Java-Umfeld auseinandergesetzt hat, wird sich womöglich fragen, was denn mit Spring sei. Spring ist ein von SpringSource gefördertes Open-Source-Projekt. Publikationen im Java-Umfeld scheinen sich heutzutage fast immer auch mit Spring zu befassen, nicht selten auch gleich mit entsprechender Erwähnung im Titel. Anstatt zu begründen, weshalb Spring eingesetzt werden soll, muss man heute als Projektverantwortlicher wohl eher begründen, weshalb Spring nicht eingesetzt werden soll: Wir waren und sind schlicht und einfach der Meinung, dass der Preis für die Einbindung und Wartung von Spring für den für Doodle zu erwartenden kleinen Mehrwert gegenüber einer JSF- und Hibernate-Lösung ohne Spring zu gross wäre.

Entwicklungsumgebung

Die wichtigsten Werkzeuge für die Entwicklung von Doodle, abgesehen vom Java Development Kit (JDK) natürlich, sind Java- und XML-Editoren sowie Ant von der Apache Foundation. Zwar setzen auch wir integrierte Entwicklungsumgebungen (integrated development environments, IDEs) wie Eclipse und NetBeans ein, aber nutzen ihre Mächtigkeit nicht einmal annähernd aus. Wir schätzen die hohe Qualität ihrer Editoren und die Möglichkeit, eigene (nicht von den IDEs erzeugte) Ant-Skripte einzubinden und ausführen zu lassen.

Mit dem Doodle-Ant-Skript können das lokale DBMS (HSQLDB) sowie der lokale Webserver (Tomcat der Apache Foundation) gestartet und gestoppt, Doodle übersetzt und für den lokalen oder produktiven Einsatz zusammengestellt sowie die Unit-Tests durchgeführt werden. Dies alles erreicht man mit einem Skript von gerade mal 400 Zeilen, das praktisch vollständig aus einfachen Einführungsbeispielen aus der Ant-Dokumentation abgeleitet werden kann.

Leistung

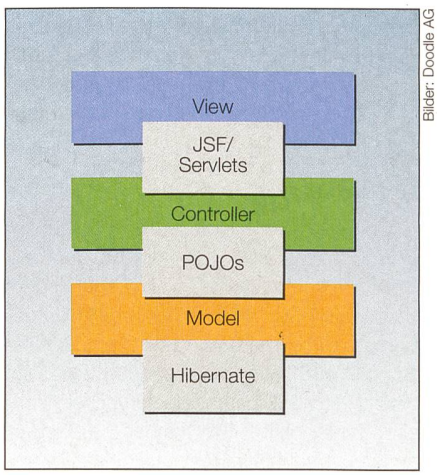
Doodle generiert und liefert pro Monat mittlerweile Millionen von Webseiten aus. Dies hat zur Folge, dass die «Müllabfuhr» (garbage collection), also das Finden und Entfernen nicht mehr genutzter Objekte, zu einem kritischen Faktor wird, da viele Objekte äusserst kurzlebig sind (nämlich einen Seitenaufruf lang) und wertvollen Speicherplatz beanspruchen.

Zwar könnte man versuchen, die Anzahl erzeugter Objekte zu minimieren, indem man trotz objektorientierter Programmiersprache auf Objektorientierung verzichtet (was gelinde gesagt seltsam anmuten würde) oder den Speicherplatz vergrössern, indem man entsprechende Investitionen in die Hardware tätigt. Doch besser früher als später sollte man sich dieser Problematik annehmen, wenn man einen grossen Webdienst betreiben möchte. Erstaunlicherweise wird sie in der Fachliteratur äusserst stiefmütterlich behandelt. Immerhin ist die entsprechende Dokumentation von Sun Microsystems hervorragend [1]. Für das bessere Verständnis sind Grundkenntnisse aus dem Gebiet der Systemsoftware (Betriebssysteme, Compiler etc.) von Vorteil.

Sicherheit

Sicherheit war bei der Entwicklung von Doodle immer eine zentrale Anforderung. Wir gehen hier exemplarisch auf einen applikationsspezifischen Aspekt ein, da die operationelle Sicherheit sinnvollerweise auf Standardbetriebssystem- und Netzwerksicherheitsmechanismen beruht.

Jede Umfrage hat einen 16-stelligen Identifikator (z.B. et6dguqr7a9fqzhw). Wer den Identifikator einer Umfrage kennt, kann damit die Umfrage einsehen (nämlich unter http://www.doodle.ch/et6dguqr7a9fqzhw), daran teilnehmen, Antworten ändern oder löschen usw. Es ist also kritisch, dass Umfrageidentifikatoren nicht einfach erraten werden können. Deshalb werden sie mit einem kryptografischen Pseudo-Zufallszahlengenerator aus einem Alphabet von 31 Zeichen erzeugt, was 31^{16} ($\approx 2^{80}$) Kombinationen zulässt. (Bei den 31 Zeichen handelt sich um die 26 Buchstaben von a bis z und die 10 Ziffern von 0 bis 9 mit Aus-



Bilder: Doodle AG

Bild 3 Die 3 Schichten von Doodle.

nahme der drei Buchstaben j, l und o sowie der zwei Ziffern 0 und 1, damit bei einer allfällig manuellen Eingabe eines Umfrageidentifikators möglichst keine Verwechslungen zwischen Buchstaben und Ziffern entstehen können.)

Wer sich von 2^{80} Kombinationen nicht beeindrucken lässt, kann aber nicht einfach ungestört Umfrageidentifikatoren durchprobieren (in einer sogenannten brute-force attack). Sollten nämlich von derselben IP-Adresse aus in kurzer Zeit verschiedene ungültige Umfrageidentifikatoren abgefragt werden, verzögert der Server die Antwort künstlich, was aus Sicht des Angreifers einer massiven Vergrösserung des Suchraums gleichkommt. Allerdings wird die IP-Adresse nicht ganz geblockt, da es durchaus sein könnte, dass der Angreifer sich die IP-Adresse mit harmlosen Benutzern teilt. Und sollte immer derselbe ungültige Umfrageidentifikator abgefragt werden, verzögert der Server die Antwort nicht – dann handelt es sich wohl eher um einen Benutzer, der erst nach dem x-ten Versuch akzeptiert, dass sein Umfrageidentifikator ungültig ist.

Résumé

Engineering Doodle

Comment programmer un service web à l'heure actuelle. Doodle permet de retrouver les rendez-vous pour réunions d'affaires, soupers entre amis ou diverses autres occasions. Il s'agit d'un service web gratuit et très populaire – qui, notons-le bien, vient de Suisse. Deux de ses caractéristiques sont sa simplicité et le fait que l'utilisateur n'a pas besoin de s'enregistrer. Et si ce service ne paraît au premier coup d'œil pas plus complexe qu'un exemple tiré d'un livre d'enseignement sur la programmation web, les exigences les plus récentes en matière de sécurité et d'ingénierie de logiciel sont satisfaites à l'arrière-plan.

Referenzen

[1] http://java.sun.com/javase/technologies/hotspot/gc/index.jsp

Angaben zu den Autoren

Dr. ***Paul E. Sevinç***, dipl. El.-Ing. ETH, ist CTO der Doodle AG. Er hat mehrere Jahre Industrieerfahrung als Software Engineer. Vor der Gründung der Inturico Engineering GmbH (die in der Doodle AG aufgegangen ist) hat er zuletzt an der ETH Zürich in Informationssicherheit promoviert.
Doodle AG, 8005 Zürich, pes@doodle.ch

Michael Näf, dipl. Informatik-Ing. ETH, ist CEO der Doodle AG. Er hat mehrere Jahre Industrieerfahrung als Security Engineer und ist Experte in Informatikdidaktik. Vor der Gründung der Inturico Engineering GmbH hat er zuletzt an der ETH Zürich Informationssicherheit gelehrt.
Doodle AG, 8005 Zürich, mn@doodle.ch