

## Frühjahrsputz beim Web Application Framework Grails

# Grails 3.0

Das Webframework Grails präsentiert sich nach einer gründlichen Überarbeitung seit Ende März in der dritten Major-Version als Grails 3.0. Mit Grails werden basierend auf der dynamisch typisierten Programmiersprache Groovy und einem sehr flexiblen Framework Webanwendungen entwickelt. Durch den Einsatz von Plug-ins können leicht weitere Features und Bausteine ergänzt werden. Das Framework Grails blickt bereits auf fast zehn Jahre aktive Weiterentwicklung, interessante Innovationen sowie Adaption von bewährten Konzepten aus anderen Webframeworks zurück.

von Stefan Glase

Mit dem Release der Version Grails 3.0 wurde der technische Unterbau von Grails rundum erneuert. So stehen mit der Aktualisierung auf Groovy 2.4 nun auch die komplette Funktionalität und die aktuelle Weiterentwicklung von Groovy in Grails-Projekten zur Verfügung. Zusammen mit dem Spring-Framework in der Version 4.1 und Spring Boot in der Version 1.2 lassen sich Grails-Webapplikationen nicht nur als WAR-Dateien bauen und in einem Servlet-Container in Betrieb nehmen, sondern auch als eigenständige JAR-Dateien erzeugen, die als notwendigen Container einen Tomcat-, Jetty- oder Undertow-Container direkt mitbringen. Damit ist Grails nun auch hervorragend als Entwicklungsumgebung von selbstständig lauffähigen Microservices geeignet.

Schon lange geplant und mit Grails 3.0 auch angegangen wurde die Ablösung des eingesetzten Build-Tools Gant. Hierbei handelt es sich um eine auf der Programmiersprache Groovy aufsetzende Erweiterung für Ant. Durch den Schwenk auf das mittlerweile auch in der Java-Welt sehr bekannte und häufig eingesetzte Build-Tool Gradle kann man in den bewährten Entwicklungsumgebungen von einer sehr guten Toolunterstützung profitieren. Durch den bereits von Gradle bekannten Cachemechanismus für wiederholte Builds profitiert jetzt auch Grails von deutlich kürzeren und schnelleren Builds.

Nach dem Austausch von Gant wurde auch das Grails Command Line Interface überarbeitet und durch ein mit Gradle integriertes, interaktives Kommandozeilenwerkzeug ersetzt. Dieses Tool nutzt die ebenfalls neu eingeführten Applikationsprofile (Application Profiles) und kann so für jedes Profil (beispielsweise Web, Plug-in oder Microservice) die passenden Kommandozeilenbefehle anbieten und komplettieren. Wie bisher können Plug-ins auch eigene Kommandozeilenbefehle mitbringen, um selbst Code zu generieren, mit Gradle zu interagieren oder Plug-in-spezifische Entwicklungstätigkeiten zu automatisieren.

Für die Entwicklungsphase ist die automatisch in jedem Projekt generierte Application-Klasse zum direkten Starten der Anwendung aus der Entwicklungsumgebung heraus eine sehr wertvolle Neuerung. So kann die Anwendung über die Main-Methode dieser Klasse im Debug-Modus gestartet, und Laufzeitprobleme können analysiert werden.

Auch für das automatisierte Testen einer mit Grails entwickelten Anwendung bringt die neue Version einige Verbesserungen. So lassen sich auch Integrationstests und funktionale Tests direkt aus der Entwicklungsumgebung heraus starten und bedürfen nicht mehr eines Aufrufs über Gant und der damit verbundenen längeren Wartezeit bis zu einer Rückmeldung durch den Test. Schon seit einigen Versionen hat das Test- und Spezifikationsframework Spock für die Ausführung von Tests im Grails-Umfeld gesorgt und wurde häufig durch das

Browserautomatisierungswerkzeug Geb ergänzt, um funktionale Tests gegen die mit Grails entwickelte Anwendung auszuführen. Beide Werkzeuge sind seit dieser Version von Haus aus in Grails integriert und müssen nicht mehr explizit konfiguriert werden.

Auch unter der Haube hat sich im Bereich der Grails-spezifischen Artefakte (Domain-Klassen, Controller, Services) einiges geändert, denn diese nutzen jetzt die mit Groovy 2.3 eingeführten Traits. Hierbei handelt es sich um wiederverwendbare Sammlungen von Methoden und Attributen, die den Grails-Artefakten ohne die Diamond-Problematik bei Mehrfachvererbung bereitgestellt werden. Aufgrund der Tatsache, dass Traits zum Zeitpunkt der Kompilierung an den konkreten Klassen hinzugefügt werden, kann die damit bereitgestellte Funktionalität nun auch aus Java-Code heraus verwendet werden.

Auch an anderen Stellen wurden neue Konzepte eingeführt. In den vorangegangenen Versionen von Grails wurden so genannte Filter implementiert und genutzt, um die Ausführung von Controllermethoden abzufangen und mit Logik anzureichern. Mit Grails 3.0 wurde das Filter-API als veraltet (deprecated) markiert und durch das Konzept des Interceptor-API ersetzt. Dieses neue API ist dem alten Konzept in vielerlei Hinsicht überlegen und besonders durch die mögliche statische Kompilierbarkeit viel performanter als das Filter-API.

In Summe können Entwickler im Grails-Umfeld von einer Vielzahl interessanter Neuerungen profitieren, und in dieser Hinsicht lässt sich optimistisch in die Zukunft schauen. Getrübt wurde dieser Eindruck jedoch im Januar dieses Jahres. Pivotal, die Firma hinter Groovy

## Entwickler im Grails-Umfeld profitieren von vielen interessanten Neuerungen, was optimistisch in die Zukunft schauen lässt.

und Grails, hat zu diesem Zeitpunkt ihren Rückzug aus beiden Projekten angekündigt. Pünktlich zur geplanten Fertigstellung der dritten Major-Version von Grails stellte Pivotal sein Engagement ein. Hintergrund dieser Entscheidung war, dass Groovy und Grails keine strategische Rolle mehr in der aktuellen Fokussierung des Unternehmens spielen, das sich stärker auf Platform-as-a-Service-(PaaS-)Dienste und datengetriebene Produkte ausrichten möchte. Während das Groovy-Projekt noch versucht, bei der Apache Software Foundation unterzukommen, ist die Zukunft von Grails vorerst gesichert: OCI (Object Computing Inc.) wird die Finanzierung des ursprünglichen Architekten- und Entwicklerteams sicherstellen.



**Stefan Glase** ist als Solution Architect im Bereich Anwendungsentwicklung für OPITZ CONSULTING Deutschland GmbH tätig. Er beschäftigt sich seit mehreren Jahren sowohl methodisch als auch technologisch mit der Anwendungsentwicklung im agilen Umfeld. Als Autor von Fachartikeln und Sprecher auf Konferenzen teilt er mit anderen seine Begeisterung für die Softwareentwicklung.

# Java<sup>TM</sup>magazin<sup>3</sup>

## Jetzt abonnieren und **3 TOP-VORTEILE** sichern!



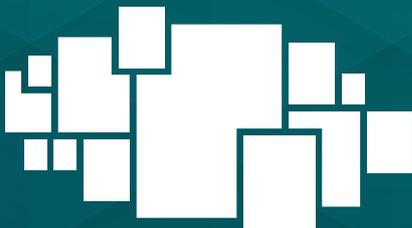
1

Alle Printausgaben  
frei Haus erhalten



2

Im entwickler.kiosk  
immer und überall  
online lesen – am  
Desktop und mobil.



3

Mit vergünstigtem  
Upgrade auf das  
gesamte Angebot  
im entwickler.kiosk  
zugreifen

Java-Magazin-Abonnement abschließen auf [www.entwickler.de](http://www.entwickler.de)