

# Java aktuell

Praxis. Wissen. Networking. Das Magazin für Entwickler  
Aus der Community – für die Community

## Java ist vielseitig



### JUnit 5

Das nächste große Release steht vor der Tür

### Ansible

Konfigurationsmanagement auch für Entwickler

### Spring Boot Starter

Komfortable Modularisierung und Konfiguration



**ijug**

Verbund

# 2016 DOAG

Konferenz + Ausstellung  
15. - 18. November in Nürnberg



Eventpartner:

[2016.doag.org](http://2016.doag.org)



2016  
**DOAG**  
Konferenz + Ausstellung





20 Anwendung im laufenden Betrieb verwalten



52 Business Process Management auf Open-Source-Basis

3 Editorial

5 Das Java-Tagebuch  
*Andreas Badelt*

8 JUnit 5  
*Stefan Birkner und Marc Philipp*

14 A Fool with a Tool is still a Fool  
*Marco Schulz*

20 Java Management Extensions  
*Philipp Buchholz*

26 Optional <Titel>  
*Dr. Frank Raiser*

30 Oracle BLOB-ZIP-Funktion für  
die Datenbank  
*Frank Hoffmann*

32 Ansible - warum Konfigurationsma-  
nagement auch für Entwickler interes-  
sant sein kann  
*Sandra Parsick*

39 Spring Boot Starter – komfortable  
Modularisierung und Konfiguration  
*Michael Simons*

44 Old school meets hype Java Swing und  
MongoDB  
*Marc Smeets*

48 REST-Architekturen erstellen  
und dokumentieren  
*Martin Walter*

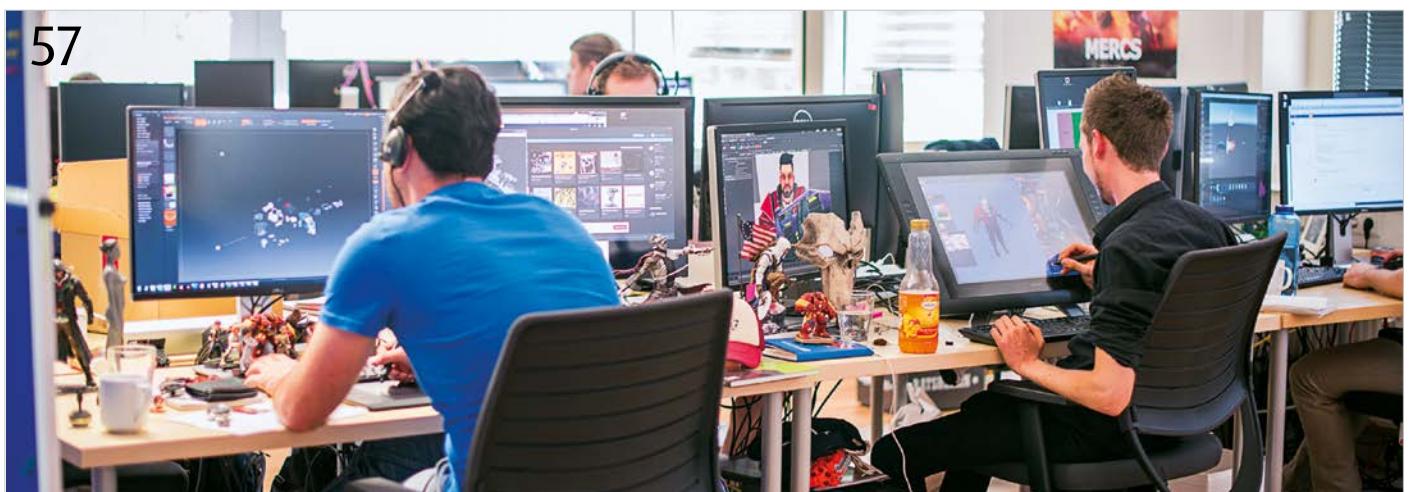
52 BPM macht Spaß!  
*Bernd Rücker*

57 Der will doch nur spielen  
*Jens Stündel*

60 Der Einspritzmotor  
*Sven Ruppert*

66 Impressum

66 Inserentenverzeichnis



57 Ein Blick hinter die Kulissen eines Spiele-Unternehmens



# Der will doch nur spielen

Jens Stündel, Goodgame Studios

*Im September veranstaltete die Java User Group Hamburg zusammen mit dem Gastgeber Goodgame Studios einen Special Day zum Thema „Gaming“. In vier hochkarätig besetzten Talks kam die Spiele-Entwicklung unter die Lupe. Der Artikel wirft einen Blick hinter die Kulissen eines Spiele-Unternehmens und zeigt, wie sich die Software-Entwicklung in der Spielebranche von der in anderen Bereichen unterscheidet.*

Software-Entwickler haben immer noch viele Vorurteile gegenüber der Spielebranche: Es werde keine ernstzunehmende Software entwickelt, dort arbeiteten nur Freaks, die nicht erwachsen werden wollen, und die Gehälter seien alle unterdurchschnittlich. Wenn dann dazu eventuell noch Eltern mit Sprüchen kommen wie

„Spielen kannst du in deiner Freizeit, mach lieber was Richtiges“, entsteht ein Bild, das Entwickler eher davon abhält, sich für diese Branche zu entscheiden. Dabei hat sich in der Spiele-Entwicklung viel getan: Online- und Mobile-Games gehören zu den am schnellsten wachsenden Märkten, es werden weltweit erfolgreiche Produkte ge-

schaffen und hochperformante Software geschrieben, und auch die Gehälter sind mittlerweile konkurrenzfähig, sodass auch erfahrene Entwickler angezogen werden.

Wenn man als Führungskraft in einem Job-Interview erfahrene Bewerber fragt, was ihnen an einer Position wichtig ist, werden in fast allen Fällen herausfordern-

de Aufgaben genannt. In der Tat stellt sich dem Entwickler in der Spielebranche eine Reihe spezieller Herausforderungen, die von außen häufig unterschätzt werden. Die Spiele-Industrie ist im Bereich der Programmierung ein besonders schnelllebiges und innovatives Gebiet. Insbesondere im „Free to play“-Segment gilt es, stets die neuesten Trends im Auge zu behalten und den Spielern in kurzen Zyklen neue, spannende Inhalte zu liefern.

Ein Spiel ist keine praktische Anwendung, die einen bestimmten Zweck erfüllt, wie beispielsweise eine Online-Banking-Software. Es soll einfach nur Spaß machen und ablenken; wenn es nicht (mehr) gefällt, kommt das nächste dran. Trends kommen und gehen – in der Unterhaltungsbranche besonders schnell. Ein erfolgreicher Spiele-Entwickler muss mit diesen Trends mithalten und idealerweise selbst Trends setzen. Dazu gehört natürlich auch die technische Basis. Neben der Marktbeobachtung, um neue Tendenzen zu erkennen, sind ständig neue Technologien zu evaluieren, Prototypen zu bauen und zu verwerfen.

Um welche Herausforderungen es in der Spieleentwicklung sonst noch geht, wird am Beispiel von Goodgame Studios dargestellt werden. Das Hamburger Unternehmen betreibt mehr als zehn Spiele weltweit, die insgesamt 300 Millionen registrierte Nutzer verzeichnen, und hat sich mit bekannten Titeln wie der App „Empire: Four Kingdoms“, deren Browser-Version „Goodgame Empire“ sowie mit „Goodgame Big Farm“ bereits international einen Namen gemacht. Mit mehr als 1.100 Mitarbeitern aus über 60 Nationen ist Goodgame Studios Deutschlands mitarbeiterstärkster Entwickler von Spiele-Software.

### Server-Programmierung mit Java

Neben dem Game-Client macht einen großen Teil der Spiele-Entwicklung die Entwicklung der Gameserver aus. Deren größte Herausforderung ist die Concurrency. Mehrere 10.000 Spieler sind in der Regel mit einem Gameserver verbunden und wollen gegen- oder miteinander spielen. Alle Aktionen der Spieler beeinflussen sich gegenseitig, dadurch ist die richtige Reihenfolge der Abarbeitung entscheidend. Merkbare Latenzen werden nicht lange toleriert und schnell mit dem Verlassen des Spiels quittiert.

Wird beispielsweise mit Locks gearbeitet, sind die richtige Locking-Strategie und ihr konsequenter Einsatz entscheidend, da Fehler in der Umsetzung sonst in teuren Deadlocks enden. Es gibt aber auch Spiele-Entwicklungsteams, die mit anderen Ansätzen wie zum Beispiel Lock-Free Programming oder dem Actor Model gute Erfahrungen gemacht haben.

Neben der Parallelität ist auch die Skalierbarkeit eines der Top-Themen. Aktuell wird bei Goodgame dabei verstärkt auf das Mesos/Marathon-Framework gesetzt. Für das Business-Intelligence-Team werden zudem große Mengen an aggregierten Daten zum Spielerverhalten gesammelt, die dazu dienen, die Spiele optimal zu justieren, um einen möglichst großen Spielspaß zu ermöglichen.

Warum wird bei Goodgame Studios überhaupt Java für die Serverseite der Spiele eingesetzt, obwohl andere Sprachen aus Sicht der Problemstellung vielleicht die naheliegende Wahl wären? Zum einen, weil sich Java als vielseitige Allzweckwaffe erwiesen hat. Alle Stärken und Schwächen sind bekannt, alle serverseitigen Anwendungsfälle sind damit umsetzbar. Außerdem lässt sich die Performance sehr gut direkt beeinflussen. Nicht zuletzt gibt es sehr gute, erfahrene Java-Entwickler – bei exotischeren Sprachen wie zum Beispiel Erlang sind diese schon schwieriger zu finden.

### Frameworks richtig eingesetzt

Die Tendenz in der allgemeinen Software-Entwicklung geht aktuell dahin, immer weiter zu abstrahieren. Man will möglichst viele Frameworks einsetzen, um die Entwicklungsgeschwindigkeit zu erhöhen und die Fehlerquote zu verringern. Bei Goodgame Studios hat man ebenfalls versucht, die Arbeit durch den Einsatz von Frameworks zu erleichtern, doch die Erfahrung hat gezeigt, dass diese oft zu sehr einschränken, zu unflexibel sind oder die Performance beeinträchtigen. So kommen beispielsweise Spring oder Hibernate nur selten zum Einsatz. Dadurch erhöht sich das Fehlerpotenzial zwar, aber da die Performance der Spiele stark profitiert, wird dieses Risiko gerne in Kauf genommen.

Hier zeigt sich dann auch das Problem auf der Suche nach geeigneten Entwicklern. Viele Bewerber sind heutzutage zwar Experten im Einsatz von Frameworks, haben aber das reine Programmieren nahezu verlernt. Dadurch scheitern sie schon

an grundlegenden Programmieraufgaben. Eine ausgewogene Mischung aus Erfahrung mit Frameworks und reiner Programmierung ist optimal.

### Lernen zu failen – failen, um zu lernen

Um Erfolge zu produzieren, ist es wichtig, Misserfolge oder Fehlschläge schnell zu erkennen. Es ist eher unwahrscheinlich, mehrere erfolgreiche Spiele direkt hintereinander zu produzieren. Zwischen zwei Erfolgen liegen immer einige Fehlversuche. Das lässt sich in einem kreativen Bereich wie der Spielebranche nicht verhindern. Umso wichtiger ist es, im Unternehmen die richtige Kultur im Umgang mit Misserfolgen zu etablieren. Je früher ein Fehlversuch erkannt wird, desto schneller kann der nächste potenzielle Hit produziert werden. Die Herausforderung ist, das Gelernte ins nächste Projekt mitzunehmen und somit die Entwicklung stets zu optimieren.

### Spieler sind die besten Tester

Im Bereich „Testing“ zeichnet sich die Software-Entwicklung in der Spielebranche durch ihre besondere Flexibilität und Agilität aus. Entgegen dem aktuellen Trend, auf Software-Entwicklungsebene zuerst zu testen und dann zu entwickeln, ist die Programmierarbeit bei Spielen weniger testgetrieben. Unit-Tests und Systemtests sind bei Backend-Services, die von allen Spielen genutzt werden, absolut sinnvoll. Doch gerade bei der Feature-Entwicklung im eigentlichen Game-Code sind sie oft hinderlich. Bei Goodgame Studios haben die erfolgreichsten Spiele beispielsweise die geringste Testabdeckung. Das ist historisch gewachsen, doch auch heutzutage können die Entwickler sowohl im Entstehungs- als auch im Instandhaltungs-Prozess eines Spiels sehr flexibel agieren, ohne stark von einem testbasierten Konzept eingeschränkt zu werden. Natürlich werden vor der Produktion eines neuen Titels die speziellen Anforderungen an das Spiel festgelegt, auf deren Basis dann ein Produkt entwickelt wird, das diese minimalen Ansprüche erfüllt. Ob es das tut, wird in mehreren Testzyklen geprüft, die jedoch die Entwicklerarbeit nicht bremsen, sondern hilfreiches Feedback für die weitere Programmierung liefern.

Nachdem die Quality-Assurance-Abteilung das Spiel gründlich getestet hat, das Produkt optimiert wurde und es marktfä-

hig ist, wird es in ausgewählten Märkten im Rahmen eines Technical Launches veröffentlicht, um die Spiele-Mechanik und technische Funktionalität zu prüfen. Nach erneuter Optimierung tritt es dann in den Soft Launch ein und wird erst dann global verfügbar, wenn die Erkenntnisse aus diesen Testphasen bestmöglich umgesetzt sind.

Auf Entwicklungs-Ebene geht es in allen Phasen darum, Fehler zu beheben, die weiterhin im Spiel auftreten können. Sobald das Spiel in einen Markt eintritt, ergeben sich Impulse aus der Spielerschaft, die ins Produkt eingearbeitet werden. Von daher ist eine schnelle Fehlererkennung und -behebung effektiver, als alle Probleme, die wahrscheinlich nie auftreten werden, schon im Vorfeld ausschließen zu wollen.

### Wöchentliche Updates

Eine Besonderheit von Online-Games sind die sehr kurzen Release-Zyklen. Bei Goodgame Studios gibt es das Ziel, für jedes aktive Spiel wöchentlich ein Update herauszubringen. Das bedeutet entweder sequenzielle Ein-Wochen-Entwicklungszyklen oder parallele, zeitversetzte Zwei-Wochen-Zyklen. Solche Parallel-Zyklen sind natürlich besonders spannend, da sie sich immer gegenseitig beeinflussen. Da muss schon der ein oder andere Konflikt gelöst werden. Agile Methoden haben sich hier als eine sinnvolle Lösung etabliert. Jedes Team findet die für sich passende Methode und entwickelt sie individuell weiter. So entstehen maßgeschneiderte Varianten von Scrum und Kanban oder auch Mischungen aus beiden – immer mit dem Ziel, die effektivste Teamorganisation zu finden.

Die Grundprinzipien sind in jedem Team vertreten: So beginnt ein Sprint mit einem Planning-Meeting. Während eines Sprints gibt es tägliche Stand-ups und am Ende werden die Ergebnisse in einem Review präsentiert und abgenommen. Besonderen Wert wird auf die Retrospektive gelegt. Die Herausforderung ist es, durch offenes Feedback und daraus folgende Verbesserungen ein optimales Umfeld für das Team zu schaffen. Bei Goodgame werden die Entwicklungsteams durch dedizierte Agile Coaches unterstützt.

### Knowledge Transfer

Es ist wichtig, immer einen regen Austausch der Entwickler zu fördern, sodass optimale Lösungen entstehen. Dadurch verbessern

sich sowohl die Code-Qualität als auch die fachliche Kompetenz der Entwickler. Die effektivste Weiterbildungsmaßnahme ist hier der Austausch untereinander, um Wissens-Silos zu vermeiden. Pair Programming oder auch Mob Programming sind gern gesehen. Code-Reviews sind fester Bestandteil des Entwicklungsprozesses. Die meisten Teams setzen auf Two-Face-Commits, es wird also kein Code freigegeben, den nicht mindestens zwei Entwickler gesehen haben.

Wenn man in kleineren, separaten Teams arbeitet, kann der Wissensaustausch über die Teamgrenzen hinweg zur Herausforderung werden. Dieser Wissensaustausch ist aber ungemein wichtig für ein effizientes Arbeiten in einem Unternehmen, etwa um den gleichen Fehler nicht mehrmals zu begehen.

Bei Goodgame Studios gibt es verschiedene Maßnahmen, um den Knowledge-Transfer zu fördern. Dazu gehören ein firmenweites Wiki ohne Lesebeschränkung, Zugang zu den Repositories aller Projekte, Communities of Practice (CoP) zu relevanten Themen sowie dedizierte Slack-Channels für den schnellen direkten Austausch. Zudem gibt es einen Pool von Architektur-Experten, der allen Teams zur Verfügung steht. Besteht in einem Team eine anspruchsvolle Architekturaufgabe, bei der es Unterstützung braucht, kann es sich einen Experten für eine begrenzte Zeit aus dem Pool holen. Das hat den Vorteil, dass der Experte sein Wissen ins Team bringt und andererseits auch das Wissen aus dem Team in andere Teams mitnehmen kann. Die Experten tauschen sich wiederum untereinander aus.

Ein weiteres beliebtes Angebot zum Wissensaustausch sind die Tech Talks. In dieser Reihe werden regelmäßig interne Vorträge auf Konferenzniveau zu aktuellen Themen veranstaltet, präsentiert von erfahrenen Mitarbeitern.

### Fazit

Die Gaming-Branche ist besser als ihr Ruf. Entwickler haben die Möglichkeit, sich mit den aktuellsten technologischen Entwicklungen zu beschäftigen und daran selbst zu wachsen. Interessante Herausforderungen gibt es genug. „Out of the box“-Denken wie auch Kreativität werden hier gefördert und gelebt, die eigenen Produkte sind weltweit eingesetzt und machen den Menschen Freude. Wer also gerne flexibel, in schnellen Zyklen und kreativ arbeitet, ist

als Software-Entwickler in einer Gaming-Company genau richtig.

Die Zukunft hält sicherlich noch einige spannende Herausforderungen für die Spielebranche bereit. Aufgrund der Schnelllebigkeit des Markts müssen Spiele-Entwickler stets neue Technologien im Auge behalten. Wird dabei etwas Interessantes entdeckt, sollte der Einsatz evaluiert und eventuell mit einem Prototyp verifiziert werden. Zum Beispiel wird schon lange mit Augmented Reality (AR) experimentiert und der Erfolg von Spielen wie Pokémon Go bestärkt diesen Ansatz. Auch Virtual Reality (VR) wird in Zukunft eine große Rolle spielen, nicht nur in der Gaming-Branche. Eins ist sicher: Langeweile wird bei der Entwicklung nicht aufkommen.

Jens Stündel

[jstundel@goodgamestudios.com](mailto:jstundel@goodgamestudios.com)



Jens Stündel hat mehr als fünfzehn Jahre Erfahrung in der Entwicklung von Online- und Mobile-Games und dabei überwiegend Java als Server-Technologie eingesetzt. Seit fünf Jahren ist er bei Goodgame Studios als Head of Game Technology verantwortlich für Technologie-Empfehlungen, die Bereitstellung von Basismodulen und gemeinsam genutzten Services für alle Spiele, die Evaluation neuer Technologien und Methoden sowie Knowledge-Transfer über Teamgrenzen hinweg.