

inkl.

CD

Scala Bytes: Pimp my Library »14



Deutschland €8,50 Österreich €9,80 Schweiz sFr 16,80 11.2011

JAVA Mag

Java™ magazin

Java • Architekturen • Web • Agile

www.javamagazin.de

CD-INHALT



Warum so viele kluge Leute so schlechte Oberflächen entwickeln

Video von der JAX 2011

HIGHLIGHT

Apache Wicket 1.5
Apache Hadoop
JUnit 4.9

WEITERE INHALTE

- Apache ZooKeeper 3.3.3
- MyFaces Core 2.1.2
- AspectJ 1.6.11

Alle CD-Infos ab Seite 3

Geoinformationssysteme

Hüter des Ortes »57

Mobile Welten

Die neue Weltordnung »64

w-jax'11

Brandneues Programm »67

Apache Hadoop

Die neue Sicht auf Daten »24 +++ MapReduce 30

+++ HBase »39 +++ Mahout »48

+++ Interview mit Ted Dunning »46

JSF und CDI

Das neue Tutorial »68

Integration mal anders

Alternative Architekturansätze »52

JUnit-Tests

Ein Programmierer sieht rot »102



Datenträger enthält Info- und Lehrprogramme gemäß §14 JuSchG

Neuer Amazon Web Service: Ein schlüsselfertiger Cache

Wolken-Cache



Amazon baut sein Cloud-Angebot jenseits einfacher virtueller Rechner und Storage konsequent aus: Neben einer Tomcat-basierten Laufzeitumgebung (Elastic Beanstalk) und MySQL bzw. Oracle-Datenbanken (Relational Database Service, RDS) gibt es nun auch eine Beta für ein Cache: ElastiCache.

von Eberhard Wolff



ElastiCache kann genutzt werden, um Daten im Speicher zu halten, den Zugriff auf andere Systeme oder Datenbanken einzusparen und so die Performance der Anwendung zu optimieren. ElastiCache basiert auf dem Open-Source-Projekt Memcached, es wird genauso wie ein Memcached-Server angesprochen. Jede Memcached Library funktioniert auch mit ElastiCache. Die Idee dahinter ist, auf speziell dafür vorgesehenen Servern Daten abzulegen. Der Zugriff erfolgt über das Netz. Dieser Ansatz unterscheidet sich also von einem Cache, der in der Anwendung selbst angelegt wird, wie dies häufig bei O/R-Mappern wie Hibernate der Fall ist. Der Ansatz hat den Nachteil, dass der Zugriff auf den Cache über das Netzwerk erfolgt, was zu einer höheren Latenz führt. Der Vorteil ist, dass so auch sehr große Caches möglich sind, da der gesamte Speicher eines Servers oder gar eines Clusters für den Cache genutzt werden kann. Die Speicherverwaltung übernimmt der Cache, sodass die JVM und ihre Garbage Collection außen vor bleiben. Java-Umgebungen sind auf kurzlebige Objekte optimiert und nicht auf Caches mit ihren langlebigen Objekten. Bei einem Cluster kann durch Replikation der Daten auch eine hohe Verfügbarkeit des Caches erreicht werden. Memcached wird von zahlreichen großen Websites wie Twitter, YouTube oder Flickr genutzt.

ElastiCache und Memcached legen unter einem Schlüssel einen Wert ab. Unter dem Schlüssel können prinzipiell beliebige Daten liegen – also Webseiten genauso wie Daten, die sonst aus der Datenbank ausgelesen würden. Memcached bietet dabei ein recht einfaches Protokoll an, für das zahlreiche Clients [1] – auch für Java – existieren. Für Java bietet sich zum Beispiel spyMemcached [2] an.

Mit ElastiCache bietet Amazon schlüsselfertige Memcached-Installationen an. Dabei gibt es Cache Nodes mit unterschiedlich viel Speicher – von 1.3 GB bis zu 68 GB. Die Kosten richten sich nach der Größe der verwendeten Cache Nodes und werden pro Stunde abgerechnet. Durch das Clustering kann ein vorhandener Cache mit zusätzlichen Cache Nodes ergänzt werden. In der Amazon-Cloud-Infrastruktur ist das Anlegen eines neuen Rechners und damit einer neuen Cache Nodes problemlos möglich. Dieser Ansatz ist unter dem Namen „Scale Out“ bekannt, weil höherer Last durch die Nutzung von mehr

Rechnern begegnet wird. Amazon gibt Memcached-Konfigurationen vor, die auf die Knotengröße angepasst sind, aber eigene Anpassungen sind mit Cache Parameter Groups auch möglich. Um Ausfallsicherheit zu gewährleisten oder mit großen Datenmengen umzugehen, können Cache Nodes zu Cache Clusters zusammengefasst werden. Amazon teilt seine Cloud in Regions wie US-East oder EU-West auf. Zurzeit ist ElastiCache als Beta und nur in EU-East nutzbar. Innerhalb dieser Region gibt es Availability Zones (getrennte Rechenzentren). Sie sind so organisiert, dass der Ausfall einer Availability Zone keine Konsequenzen für anderen Zonen hat. Ein Cache Cluster ist in einer Availability Zone. Sollte die für die benötigte Verfügbarkeit nicht ausreichen, können mehrere Cache Cluster in unterschiedlichen Availability Zones genutzt werden. Der Zugriff auf den Cache kann durch „Security Groups“ eingeschränkt werden. Sie sind vergleichbar mit Firewall-Konfigurationen und erlauben den Zugriff auf die Server nur von bestimmten IP-Adressen auf bestimmte Ports. Vorteil von ElastiCache: Zunächst muss der Kunde sich nicht um die Installation von Memcached kümmern. Weitere Vorteile:

- Der Ausfall einer Cache Node wird automatisch festgestellt und ein neuer Rechner wird gestartet, der auch automatisch in den Cluster aufgenommen wird.
- Updates und Patches werden von Amazon eingespielt. Für die Installation wird ein von Kunden definiertes Zeitfenster genutzt.
- Mithilfe des Cloud-Watch-Dienstes können die Cache Nodes überwacht werden. Dabei werden auch Daten gemessen, die spezifisch für den Cache sind wie beispielsweise Cache Hits.

Damit setzt Amazon den Ansatz fort, sein Angebot im Bereich IaaS mit virtuellen Rechnern und Storage um komplexere Dienste zu ergänzen. Dazu zählt z. B. auch der Relational Database Service (RDS), bei dem man Oracle- oder MySQL-Server mieten kann und ähnliche Vorteile hat wie bei ElastiCache. Ebenfalls gibt es mit Elastic Beanstalk ein PaaS, das auf Tomcat basiert und auf das man sehr einfach Java-Anwendungen deployen kann. Wie die anderen Amazon-Dienste auch kann der Nutzer ElastiCache auf unterschiedliche Weise steuern:

- Über die Webschnittstelle werden Caches, Cache Nodes und Cache Cluster erzeugt und konfiguriert.
- Dieselben Möglichkeiten bieten die Kommandozeilenwerkzeuge. Dadurch können Abläufe automatisiert werden.
- Wer mehr Flexibilität benötigt, kann auf die SOAP-Web-Services-Schnittstelle zugreifen und zur Automatisierung eine beliebige Programmiersprache nutzen.

So kann ElastiCache manuell administriert oder Prozesse automatisiert werden. Die Schnittstelle des Caches selbst bietet die Möglichkeit, Daten unter einem bestimmten Schlüssel abzulegen und abzurufen. Dabei ist es auch möglich, festzulegen, wie lange die Daten im Cache vorgehalten werden sollen. Außerdem kann ein numerischer atomar-Wert erhöht werden. So können netzwerkweite Zähler eingerichtet werden. Da die Daten im Speicher vorgehalten werden, ist der Zugriff performant möglich. Der Cache unterstützt optimistische Concurrency Control: Beim Ändern eines Wertes kann der aktuelle Wert mit einem vorher gelesenen Wert verglichen werden. Wenn in der Zwischenzeit jemand den Wert geändert hat, bekommt man eine Fehlernachricht. Für eine Einführung in ElastiCache bietet sich [3] an.

Fazit

ElastiCache bietet Memcached schlüsselfertig in der Cloud an. Damit fügt Amazon seinen Angeboten einen weiteren interessanten Dienst hinzu, der für die Performance der Anwendungen zentral sein kann und sich in zahlreichen Projekten bewährt hat. Solche In-Memory-Ansätze sind in der Cloud besonders gut nutzbar, da man dort leicht neue Server zu einem Cache Cluster hinzufügen kann. Für den Entwickler ist der Cache im Wesentlichen eine große Map – die Herausforderung ist, welche Daten man in dem Cache hält, um so einen möglichst großen Vorteil daraus zu ziehen.



Eberhard Wolff (Twitter: @ewolff) arbeitet als Architecture and Technology Manager für die adesso AG in Berlin. Er ist Java Champion, Autor einiger Fachbücher und regelmäßiger Sprecher auf verschiedenen Konferenzen. Sein Fokus liegt auf Java, Spring und Cloud-Technologien.

Links & Literatur

- [1] <http://code.google.com/p/memcached/wiki/Clients>
- [2] <http://code.google.com/p/spymemcached/>
- [3] <http://docs.amazonwebservices.com/AmazonElastiCache/latest/GettingStartedGuide/>

Anzeige