

Cloud Programming

Stefan Wurzinger

6. Juni 2013

1 Einführung

Dieses Dokument bietet eine Einführung in die Cloud Programmierung. Zum leichteren Verständnis geschieht dies in zwei Stufen. Kapitel 2 bespricht das Cloud Computing Modell und Kapitel 3 gibt, anhand eines Beispiels, einen Einblick in das Programmieren in der Cloud.

Um die Lesbarkeit zu erleichtern, wird auf die zusätzliche Formulierung der weiblichen Form verzichtet. Die männliche Form soll als geschlechtsneutral verstanden werden.

2 Cloud Computing

2.1 Was ist Cloud Computing?

Cloud Computing beschreibt ein relativ neues Modell in der Informationstechnik. Die US-amerikanische Standardisierungsstelle NIST (National Institute of Standards and Technology) definiert Cloud Computing folgendermaßen:

Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. [1]

2.2 Das Cloud Konzept

Durch die rasante Entwicklung des Internets und die immer besser werdende Performanz von Servern (bessere Rechenleistung, mehr Speicherplatz) greift das Cloud Konzept die Idee auf, Applikationen über das Internet zur Verfügung zu stellen und an einem zentralen Ort zu verwalten. Die Daten und Applikationen befinden sich somit nicht mehr auf den privaten, lokalen Rechnern, sondern in der Cloud.

2.3 Servicemodelle

Cloud Computing kann in drei verschiedene Kategorien von Servicemodellen unterteilt werden.

2.3.1 Infrastructure as a Service (IaaS)

IaaS ermöglicht es dem Cloud-Kunden IT-Ressourcen, wie zum Beispiel Rechenleistung und Speicherplatz, anzumieten. Der Kunde kann mit Hilfe dieses Service seinen eigenen virtuellen Computer-Cluster konfigurieren und betreiben.

2.3.2 Platform as a Service (PaaS)

Ein PaaS-Provider stellt eine komplette Infrastruktur mit Zugang zu standardisierten Schnittstellen, zum Beispiel Datenbanken, E-Mail etc., zur Verfügung. Der Kunde kann seine gewünschten Anwendungen installieren, hat jedoch keinen Zugriff auf das Betriebssystem und die darunterliegende Hardware. Cloud-Entwickler nutzen diesen Service oft als Grundlage für die Erstellung von neuen Applikationen.

2.3.3 Software as a Service (SaaS)

Dieser Service erlaubt es, Anwendungen über die Cloud zu beziehen. Ein Beispiel für eine solche Anwendung ist Google Drive ¹.

2.4 Arten von Clouds

Ergänzend soll erwähnt werden, dass es drei Arten von Clouds gibt: Public Cloud, Private Cloud und Hybrid Cloud. Der interessierte Leser wird an [2] und [3] verwiesen.

2.5 Cloud für Entwickler

Wenn sich ein Entwickler dafür entscheidet, seine Applikation über die Cloud zur Verfügung zu stellen, kommt er in den Genuss vieler Vorteile. Der Entwickler kann sich auf seine wesentliche Arbeit konzentrieren und muss sich nicht darum kümmern, wie die Software installiert wird, die Software und die Hardware gewartet wird und ob die Daten sicher und verschlüsselt abgelegt werden. Dies alles übernimmt der Service-Provider. Durch die serviceorientierte Architektur von Cloud Computing wird es dem Entwickler ermöglicht, sein System optimal zu skalieren. Immer wieder kommt es vor, dass während der Lebensdauer einer Software, Ressourcenengpässe entstehen. Bei bisherigen, traditionellen Systemen musste man die Hardware selbst aufstocken, um die Ressourcenengpässe zu beseitigen. Dies führte im ungünstigsten Falle zu einer kompletten Umstrukturierung der Serverarchitektur. In Zeiten von Cloud Computing gehört dies der Vergangenheit an. Da Cloud-Produkte als Service angeboten werden, kann das System ohne großen Aufwand skaliert werden, indem man zusätzliche Ressourcen vom Service-Provider erwirbt. Weitere Vorteile ergeben sich durch die zentrale Verwaltung der Applikation: Es muss keine Zeit in die Entwicklung der Anwendungsportabilität (Lauffähigkeit auf mehreren Betriebssystemen) investiert werden, die Installation auf diversen Endgeräten entfällt, aber auch Updates müssen nur auf dem gemieteten Server eingespielt werden.

¹<https://drive.google.com/> - abgerufen am 02.06.2013

2.6 Welche Arten von Applikationen sollen in der Cloud platziert werden?

Grundsätzlich können alle Applikationen in der Cloud eingebettet werden, jedoch ist die Cloud der ideale Ort für folgende Arten von Anwendungen:

- Gemeinsam genutzte Applikationen
- Services
- Applikationen, die eine hohe Rechenleistung benötigen

3 Programmieren in der Cloud

3.1 Cloud Computing Systeme

Um eine Applikation in der Cloud zu platzieren, wird ein Cloud Computing System benötigt. Die Google App Engine ² ist ein solches System. Sie bietet viele Services an, die dem Entwickler helfen, eine Anwendung leichter und schneller zu implementieren. Zu diesen Services gehören unter anderem Authentifizierungs-, Datenbank- und E-Mail-Schnittstellen.

3.2 Entwicklung einer Applikation

Wie in Sektion 3.1 schon erwähnt, ermöglicht das Google App Engine System (GAE) in der Cloud zu programmieren. Listing 1 wurde mit Hilfe von GAE und Java geschrieben. GAE Java Anwendungen nutzen die Java Servlet Technologie ³, um mit dem Webserver zu interagieren. Das Programm aus Listing 1 erlaubt authentifizierten Benutzern eine Nachricht auf dem System zu hinterlassen, ähnlich einem Forum oder Gästebuch. Es werden zwei Services der GAE verwendet, der Authentifizierungsservice (UserService) und der Datenbankservice (DatastoreService). Mittels dem Authentifizierungsservice können sich Benutzer, die ein Google-Konto haben, anmelden. Die Anmeldung erfolgt über eine Benutzeroberfläche die vom Service bereitgestellt wird. Der Datenbankservice kümmert sich um die Zugriffe auf die Datenbank. Des Weiteren werden auch Replikationen und Lastausgleiche von diesem Service durchgeführt. Wie wir sehen können, sind diese Services sehr hilfreich für die Entwicklung von neuen Anwendungen, zumal heutzutage fast jede Anwendung eine Authentifizierung und eine Datenbank benötigt. Es sei noch erwähnt, dass auch andere Datenbankservices zur Auswahl stehen. Für weitere Informationen siehe [4].

²<https://developers.google.com/appengine/> - abgerufen am 02.06.2013

³<http://www.oracle.com/technetwork/java/javaee/servlet/index.html> - abgerufen am 02.06.2013

Listing 1: MessageServlet.java

```
public class MessageServlet extends HttpServlet {
    public void doPost(HttpServletRequest req, HttpServletResponse resp) {

        UserService userService = UserServiceFactory.getUserService();
        User user = userService.getCurrentUser();

        if(user == null)
            return; // Kein Benutzer angemeldet!

        Key key = KeyFactory.createKey("Messages", "default");
        Entity message = new Entity("Message", key);
        message.setProperty("user", user);
        message.setProperty("date", new Date());
        message.setProperty("text", req.getParameter("text"));

        DatastoreService datastore = DatastoreServiceFactory.getDatastoreService();
        datastore.put(message);

        // Nachrichten anzeigen
    }
}
```

3.3 Applikation veröffentlichen

Applikationen die veröffentlicht werden sollen, müssen über das verwendete Cloud Computing System registriert werden. Nach der erfolgreichen Registrierung wird eine URL generiert, mit der auf die Applikation zugegriffen werden kann.

4 Schlussfolgerung

Die Cloud Programmierung bringt, wie wir gesehen haben, viele Vorteile mit sich. Es macht jedoch wenig Sinn, alle Applikationen, wie zum Beispiel Programme, die eine hohe Datensicherheit aufweisen müssen, in der Cloud abzulegen. In der Industrie wird Cloud Computing aus Sicherheitsgründen kaum eingesetzt, dennoch denke ich, dass wir in Zukunft noch sehr viel von dieser neuen Technologie hören werden.

Literatur

- [1] Peter Mell und Timothy Grance. The NIST Definition of Cloud Computing, 2011.
- [2] Mark C. Chu-Carroll. *Code in the Cloud - Programming Google App Engine*. The Pragmatic Programmers, LLC, Dallas, Texas, 2011.
- [3] Borku Furht und Armando Escalante (Eds.). *Handbook of Cloud Computing*. Springer, New York, 2010.
- [4] Google App Engine - Getting Started: Java. <https://developers.google.com/appengine/docs/java/gettingstarted/>. Abgerufen am 02.06.2013.