Hybrid cloud architectures for the online commerce

Georg Lackermair a, b *

a Competence Center Information Systems and Management, Munich University of Applied Sciences, Lothstraße 64, 80335 Munich, Germany
b Technical University Dresden – Faculty of Business and Economics, Chair of Business Informatics, esp. IS in Manufacturing and Commerce, 01062 Dresden, Germany

Abstract

Cloud Computing is a relatively new computing approach which promises enhanced scalability, flexibility and cost-efficiency. In practice, however, there are many uncertainties about the usage of this way of provisioning IT resources. Concerns about evolving dependencies and security issues have arisen. As online commerce is dependent on a reliable, secure online stores, it is vital to take those concerns into account. The hybrid architecture combines traditional IT service provisioning with the usage of internet-based services. This approach would help to harness the strengths of Cloud Computing, while minimizing the impacts of its potential risks. This article discusses hybrid cloud architectures with the focus on realizing shop functionalities. Advantages and drawbacks of integration for each service type in the Cloud Computing stack are given, from which possible problems of hybrid clouds are derived. The conclusions in this article provide a conceptual basis for further research and evaluation of the use of net-based services in the field of online commerce.

Keywords: Cloud Computing; E-commerce; Hybrid Cloud; Web-based Services

1. Motivation

Classic online retailers show just limited interest for the term Cloud Computing as own observations have shown. The model of IT delivery that implies developing custom and stable features with the known supplier and running the application on a dedicated data center suits well, as a shop system is perceived rather as the source of income than as a cost factor. The environment of E-commerce, however, changes rapidly and there are currently various variables that change: Competition, demand for sustainability and participation, rapid development of Cloud Computing offerings, the Social Web, etc. Cloud Computing promises to be the solution to many of those challenges. However, many concerns about transparency, data security and quality of service prevent companies from adopting this approach.

Hybrid architectures that combine traditionally provided IT services with web-based services could enable to harness the advantages of this computing paradigm, while minimizing the risks. This paper describes various scenarios of hybrid architectures from a B2C retailers point of view.

First, a comprehensive introduction of E-commerce systems and the term Cloud Computing is given. The main part of the paper discusses advantages as well as disadvantages of each architectural scenario concluding in an
outlook on future research.

2. E-commerce systems

For describing the different aspects of an online commerce systems, this paper uses the work of Laudon and Traver [1, p. 1-9], who define E-commerce as „digitally enabled commercial transactions between and among organizations and individuals.“ Figure 1 shows a high-level view on an E-commerce system's architecture, which is the result of several case studies of existing online retailers both in B2B and B2C commerce.

![Fig. 1: A high-level system architecture of an online retailer.](image)

An E-commerce system’s architecture consists of several layers. Logistic-related systems handle the flow of physical goods through the organization. The E-commerce backend manages catalogue data and provides transaction processing functionality. An online shop is the web-based frontend for customers. A shop consists of the primary components catalogue, cart and check-out. In practice, there are various secondary functionalities, e.g. user tracking, online recommendation, support, social features, integrated in an online store.

3. Cloud Computing

Before discussing scenarios about the integration of this computing paradigm, definitions and a short introductions to the terms used are given. In scientific literature, many different definitions of Cloud Computing can be found. This paper mainly bases on the approach of Armbrust et al. who define this IT delivery model as follows: „Cloud Computing refers to both the applications delivered as services over the Internet and the hardware and systems software in the datacenters that provide those services. The services themselves have long been referred to as Software as a Service (SaaS), so we use that term. The datacenter hardware and software is what we will call a Cloud. When a Cloud is made available in a pay-as-you-go manner to the public, we call it a Public Cloud; the service being sold is Utility Computing. Current examples of public Utility Computing include Amazon Web Services, Google AppEngine, and Microsoft Azure. We use the term Private Cloud to refer to internal datacenters of a business or other organization that are not made available to the public. Thus, Cloud Computing is the sum of SaaS and Utility Computing, but does not normally include Private Clouds.“ (Armbrust et al. [2, p. 4])

Providers apply online ordering and payment via browser-based applications for selling Utility Computing and Application Service Providing (ASP). Hence, a very important aspect in Cloud Computing is E-commerce applied to the above-mentioned services. Other works introduce the service types infrastructure, platform and software for cloud-based services ([3], [4]). The term Everything as a Service (EaaS or XaaS) aggregates Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS) as described in Lenk et al. [5]. The

---

8 These observations were made in frame of discussions with four E-commerce experts, both in B2B and B2C retail. Besides that, own experience in E-commerce projects support these conclusions.
author generally prefers to use the definition of Armbrust et al. In the main part of this paper, however, for clear distinction of the usage of different service types, layers similarly to the XaaS approach are introduced. SaaS in general is Application Service Providing, which is made publicly available and can be purchased easily on the web. A definition of ASP can be found in Factor [6, p. 8]. The duration of the contract distinguishes SaaS and ASP: while in ASP it usually is more than one year, SaaS providers offer contracts with significantly shorter durations. The software serviced can offer various interfaces towards users: a web frontend for human interaction or interfaces for application integration. Such interfaces are commonly realized as HTTP, WebServices or RESTful Services (see W3C [7], Fielding [8]).

Cloud-based infrastructure provides access to virtualized hardware located on the Internet. According to Armbrust et al. the service that is provided is Utility Computing. The most popular IaaS provider is Amazon EC2. On the web interface a user can rent a virtual machine (VM) and select from a range of operating systems (OS) an image that will be installed inside the VM. The service’s web interface also provides management functionalities for a customer’s machines.

Net-based application platforms offer an OS and common services, e.g. application server, or database server. The user typically has no access to the platform, but deploys an application on a web-based frontend, which provides also administration functionalities. The providers commonly offer payment models that are based on fixed quotas of resource usage, e.g. disk space, number of requests. Examples for such platform providers are Google App Engine (GAE), Force.com or Heroku. As such platforms are managed, a part of management and maintenance shifts to the platform provider, which results in reduced complexity in operation. Besides that, a variable payment model offers calculable operation costs. Another approach is the so called "OpenSocial" approach, which allows deploying gadgets on services of the social web, e.g. Facebook, MySpace or iGoogle. This type of PaaS is an approach to standardize the programming interface of Social Networking Services (SNS) and contains a JavaScript API for developing gadgets and a RESTful server-interface for accessing the social network.

On the application layer, software-based services are obtained via Internet. There are many services for different purposes offered: general applications e.g. word processors, ERP, collaboration tools, etc are accessed by a simple web browser. Specialized services e.g. online payment, etc are integrated in enterprise systems.

On top of those three layers, Lenk et al. add the human layer to the cloud computing stack [5]. This term refers to the acquisition of human resources with means of cloud-based participation services and is often called Human as a Service (HaaS) or Crowdsourcing. Reichwald and Piller mention development of Open Source Software (OSS) or Spreadshirt as popular examples for crowdsourcing [9].

4. Hybrid Cloud Architectures for E-commerce

This section describes hybrid cloud architectures for each service type of the Cloud stack. In terms of Armbrust et al.’s distinction of Public and Private Clouds, a hybrid architecture combines services from both types of provisioning in a single information system. The scope of this paper is to deliver classic shopping functionality.

Fig. 2: Integration of different service types in an online shop
An overview over hybrid integration of the various service types is depicted in Figure 2. Details about the peculiarities of integration are discussed for each service type in the following. The human layer of Cloud Computing is not subject of this discussion, as it still is relatively unexplored in context of online retail.

4.1. Infrastructure

The idea of a partial integration of net-based infrastructure in an online shop is the result of own previous research. A detailed discussion of the architectural sketch is published in [10]. The data-centric catalogue component is distributed in net-based data-centers to support application scaling in peak situations. The non-critical catalogue data can be distributed in the cloud, for accessing personal data and for processing transactions, however, the backend needs to be accessed. Our cost-based evaluation showed, that this architecture can reduce costs of operation, due to a more accurate provisioning of IT services (see Armbrust et al. [2]).

For a system that is accessed mainly in a read-only manner, the additional distribution of cached content discharges the backend database. The danger of evolving dependencies is relatively low with this approach as most vendors allow the usage of the most common operating system platforms for installation in their virtual machines. Furthermore, our approach is limiting distribution of data to web-based infrastructure to non-critical catalogue data which reduces the danger of data lock-in.

Potential drawbacks are uncertainties about the quality of service – the network connection between the target customers and the IaaS provider might be slow, and, thus, it might be hard to predict the latency of requests. Problems of clustering such as session failover become even more challenging in such a strongly distributed environment, since the distribution over the internet results in longer round trip times. Another disadvantage is, that both the backend and the shop require several implementation changes for a secure integration: An interface for backend access and a mechanism for integration of cloud-based instances in the load-balancing need to be provided. According to Zhang et al. [11] the database is the bottleneck of E-commerce applications. A further distribution of business logic could temporarily increase the workload on the backend systems on startup. Thus, an efficient mechanism for content distribution, e.g. distributed caching, needs to be implemented.

4.2. Platform

The use of net-based application platforms holds the danger of a technological lock-in. This contrasts completely to the traditional web-application hosting, which is based on the LAMP-Stack\(^2\). As there are many web hoster that provide a LAMP environment, this configuration offers high portability. Among the evolving cloud-based platforms, however, an equivalent standard environment is not yet established. The platform providers offer different programming languages and some require programming an application against a proprietary API, which results in high switching costs (see Shapiro and Varian [12, p. 106]).

A first examination of the platform mentioned in paragraph 3 supports the picture of heterogeneity. The choice of languages, development tools and proprietary API prevents portability. Besides that, no serious standard has been proposed until now, which is quite different to the OpenSocial approach mentioned above, which links various social networking platforms. Although recently the term Open Cloud was shaped (which stands for openness and standardization among Cloud Computing providers) and although various organizations have already agreed on the need of standardization, there is still no concrete specification.\(^3\)

Own observations showed that an online shop’s core components are feature stable. This means that the development of those components is dominated by incrementally enhancing and improving existing functionalities and adopting new features slowly. Thus, the major advantage of rapid application development and deployment does not matter for this part of the store – in combination with a possible platform lock-in, it PaaS does not appear to be appealing for existing shop applications. However, online retailers might demand ad-hoc, temporary features, which are needed e.g. for marketing campaigns, introduction of new products, etc. A hybrid architecture combines

---

\(^2\) LAMP is an acronym for Linux, Apache, MySQL and PHP/Python

\(^3\) See Open Cloud Consortium (http://opencloudconsortium.org/), Open Cloud Manifesto (http://www.opencloudmanifesto.org/)
the stable development and deployment model of a shop's core with the rapid application development and deployment of a web-based platform. Such a model would also allow to test the acceptance of experimental features before integrating them into the core system.

4.3. Application

Hybrid architectures on the application layer are already quite common, since many web stores already use net-based user tracking, or social software services for corporate communication and marketing purposes. A very important process in B2C E-commerce, the payment process, is facilitated by online payment services, e.g. PayPal, Amazon Payments, Google Checkout, PaySimple.

Reasons for using Software as a Service are cost predictability and immediate operational readiness. Due to pricing models on use-basis or – in case of payment services – on transaction-basis, this IT delivery model is suitable for projects that are provided with a low budget as costs are variable and expenses incur on creation of corresponding revenue. SaaS also supports entering new markets in the online commerce: Different payment services offer access to certain, partially disjoint sets of online consumers. Schneider [13, p. 511] forecasts a growth of the share of electronic payment from 6% in 2010 to 15% in 2020 in the U.S. consumer market. Corporate communication with social software support the market-oriented argument. Integrating net-based software can help to increase the acceptance of an online shop.

A disadvantage of SaaS is, that the customer can not negotiate SLAs individually – the provider offers terms and the consumer accepts those. As a provider and a customer may rely on different jurisdictions this may lead to problems (see Hayes [14]). As a service provider is both an economically and legally independent company, the retailer's possibility to influence continuity or development of the service is limited. On technical level, the problems of Distributed Systems apply to the use of Web Services, i.e. latency of the underlying transport and partial failure (see Waldo et al. [15]).

5. Conclusions

This paper provides a comprehensive introduction to Cloud Computing. The main part discusses scenarios for a hybrid integration of net-based infrastructure, platform, software and collaboration from a retailers' point of view. With partially running the read-only parts of a shop system on cloud-based infrastructure, a shops ability to scale up as well as to scale down would be improved. Therefore, overall costs of ownership can be reduced. Such a distribution, however, would require substantial changes in the application, e.g. concerning failover or caching.

Net-based platforms could be harnessed to combine an agile development of experimental features with the incremental development of the stable core features. The light-weight programming models enable rapid development and, thus, PaaS can be used to test experimental features – probably developed in cooperation with academia. The examples of social networking platforms show that net-based platforms offer also access to services and data, which enables new functionalities that hardly can be built by one self. A disadvantage is still the lacking portability of applications.

The integration of cloud-based services in various domains is already quite common: payment services and user tracking are widely used in online commerce. Particularly new payment services with their user base help retailers to enter new markets. Concerns about data privacy and security and the dependency on connectivity are major disadvantages. As the provider is an independent company, issues connected to the legal basis and standardized SLAs could arise. Besides that, a retailer needs to evaluate the economic stability of the supplier.

Crowdsourcing on top of the Cloud Computing stack is relatively new to online retailers, although Reichwald and Piller mentions various successful examples, e.g. Spreadshirt [9, p. 51]. There are various different forms of integration of consumers in the value creation, whereas it is still unclear how the Social Web can be used for collaboration with consumers. Studies about switching barriers in the internet conclude, that perceived value generates a high customer loyalty [16], [17] which suggests, that satisfied and loyal consumers can be motivated for collaboration. Affiliate marketing as a widely applied form of crowdsourcing supports this conclusion (see [18], p. 13). In the author's opinion is the human layer a promising field of future E-commerce research.
References