



What can you do with AWS?

By Michael Wittig and Andreas Wittig

You are able to run any application on AWS by using one or a combination of different services. The four examples we present in this article will give you an idea of what you can do with AWS.

This article is excerpted from [Amazon Web Services in Action](#). Save 39% on Amazon Web Services in Action with code 15dzamia at [manning.com](#).

What can you do with Amazon Web Services? You are able to run any application on AWS by using one or a combination of different services. The four examples we present in this article will give you an idea of what you can do with AWS.

Hosting a web shop

When you are hosting a web shop on-premise, you would typically rent a server in a data center. On this server you run a web server, handling the requests from your users and a database storing your products and orders. You can run this exact same setup in AWS as figure 1 shows.

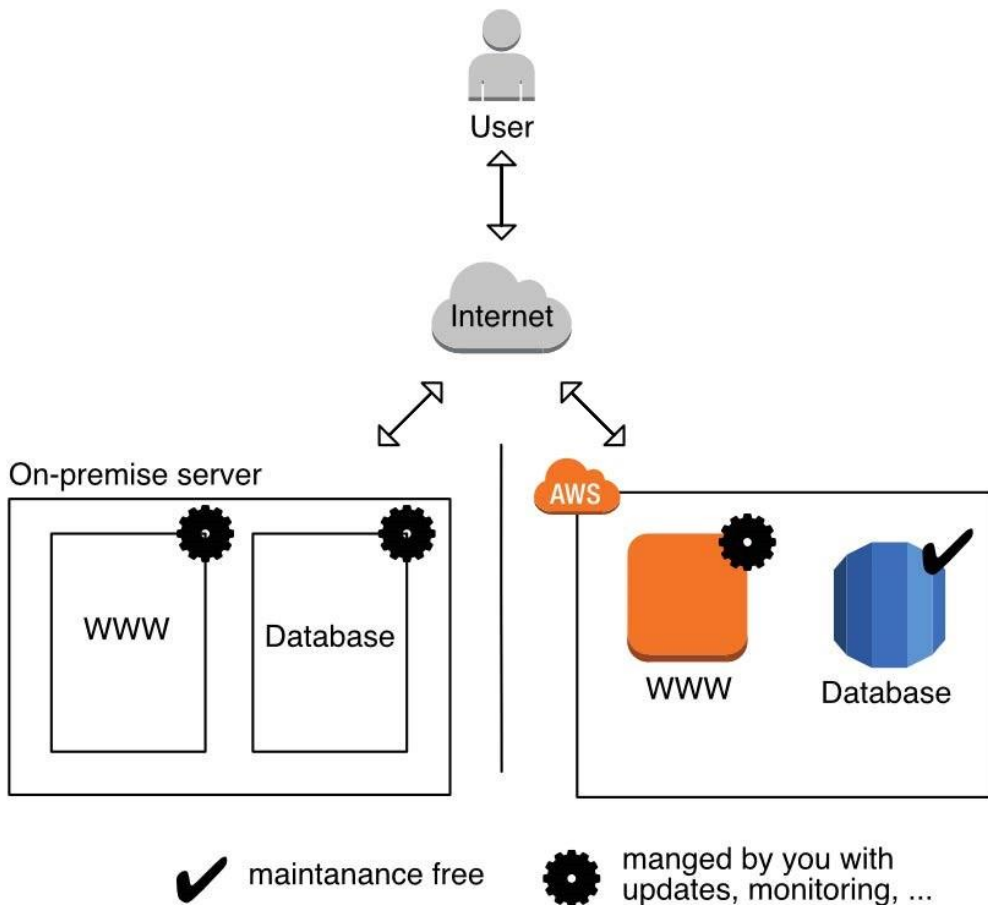


Figure 1 Comparison of how to run a web shop on-premise and on AWS

You might ask yourself: Why should I do that? Only because I don't have to care about the database any longer? But look at how the setup could be improved to really benefit from AWS in three steps.

1. Your web shop consists of dynamic content like your products and their prices and static content like your logo. By splitting dynamic and static content you can reduce the load for your web servers and improve performance by delivering the static content over a Content Delivery Network (CDN).

For source code, sample chapters, the Online Author Forum, and other resources, go to <http://www.manning.com/wittig/>

2. You are able to use maintenance-free services like a database, an object store or a DNS system on AWS. This frees you from managing these parts of your system, decreases operational costs and improves quality.
3. The application running your web shop can be installed on virtual servers. You are able to split the capacity of your old on-premise server into multiple smaller virtual servers at no extra cost. If one of these virtual servers fails, your load balancer will send the requests from your users to the other virtual servers. Doing so will improve the reliability of your web shop.

Figure 2 compares the on-premise setup with the enhanced AWS setup.

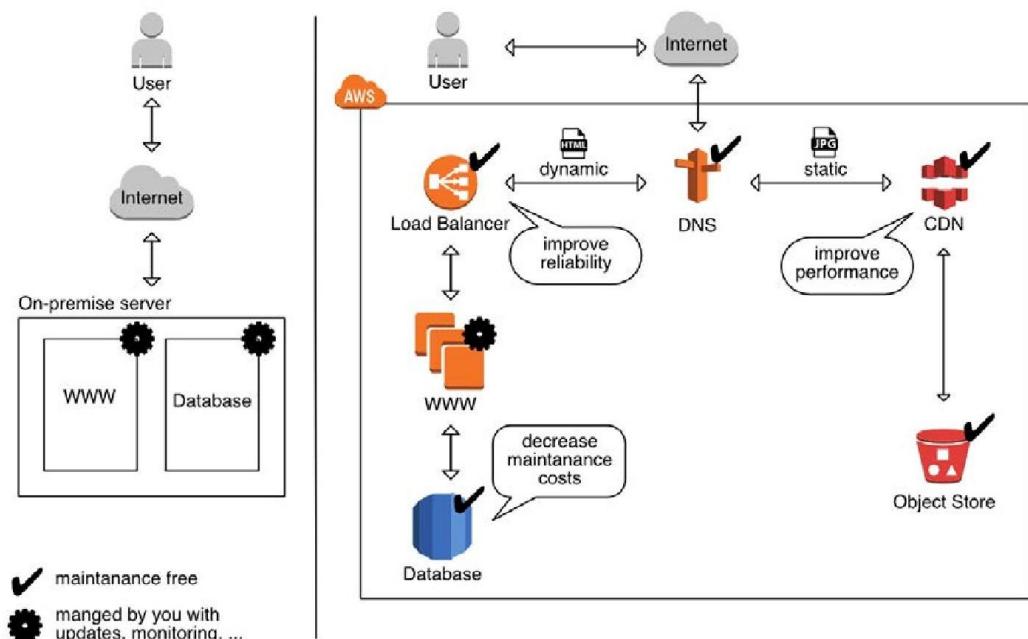


Figure 2 Comparison of how to run a web shop on-premise and on AWS and benefit from the cloud

Any web application can be transferred to AWS without any constraints. You can optionally use a SQL database or an object store to get the most out of the cloud.

It's possible to transfer any application to AWS, even an enterprise application.

Running a JavaEE application in your private network

It is possible to run an enterprise application on AWS. To do so you need to define a virtual network in the cloud and connect it to your corporate network through a VPN connection.

For source code, sample chapters, the Online Author Forum, and other resources, go to

<http://www.manning.com/wittig/>

Controlling access and protecting mission-critical data can be achieved with subnets and access control lists for the communication between them. You can control traffic to the Internet by the use of NAT and firewalls. Application servers can be installed on virtual machines to run your JEE application. If you like, you can store your data in a SQL database service (e.g., Oracle Database Enterprise Edition or Microsoft SQL Server EE). Figure 3 illustrates this scenario.

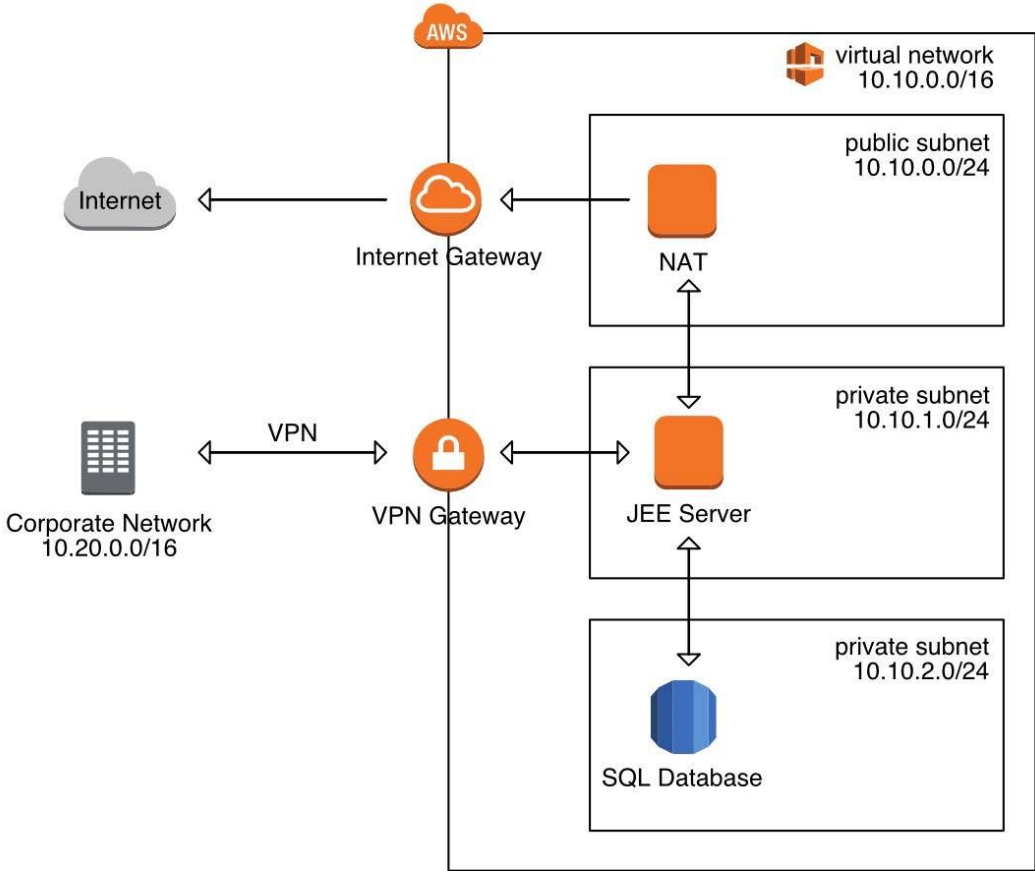


Figure 3 Running a JEE application with enterprise networking on AWS

If you have to deal with huge amounts of data to meet legal or business archival requirements, the next use case will show you how you can handle this with AWS.

Meeting legal and business data archival requirements

You can find a file server for sharing documents between users in every company. Storing data always creates problems.

1. You need to back up all files to prevent the loss of critical data. Usually this means you have to copy the data from your file server to another network attached storage. This means you have to buy the hardware for your file server twice. If the file server and the backup server are located close together, you will fail to meet disaster recovery requirements.
2. To meet legal and business data archival requirements, you are forced to store data for a long time. Storing data for 10 years or longer is tricky. You will need a tape deck and safe deposit box to do so.

To save costs and increase data security, you can implement this use case on AWS. Data is transferred to a high available object store. A storage gateway makes it superfluous to buy and operate network attached storage and a backup on-premise. A virtual tape deck can take over the task of archiving data for a long time. Figure 4 illustrates the implementation of this use case on AWS and compares it to an on-premise solution.

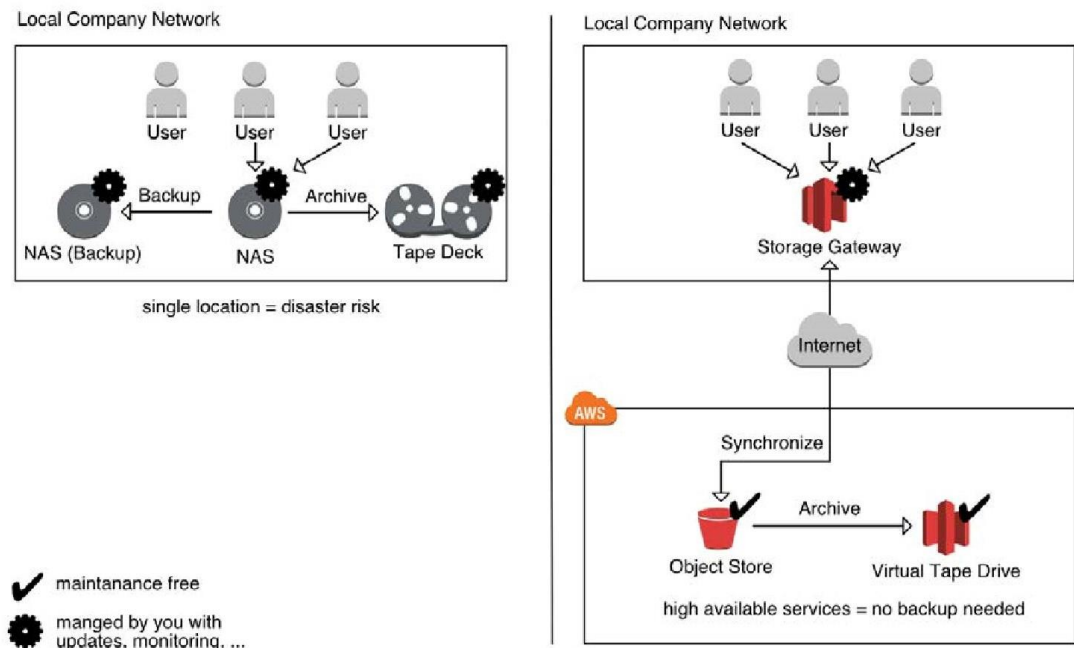


Figure 4 Backup and archive data on-premise and on AWS

For source code, sample chapters, the Online Author Forum, and other resources, go to

<http://www.manning.com/wittig/>

You can also design sophisticated fault-tolerant systems with AWS.

Implementing a fault-tolerant system architecture

Murphy's law applies to IT infrastructure: anything that can go wrong will go wrong. On AWS there are two type of services: fault tolerant services and services that can be used in a fault-tolerant way. Figure 5 shows a fault-tolerant architecture. The database service is offered with replication and failover handling. A single virtual server, used as a web server in this example, is not fault tolerant. But with the help of a load balancer and the possibility to launch multiple servers in different data centers, fault tolerance can be achieved.

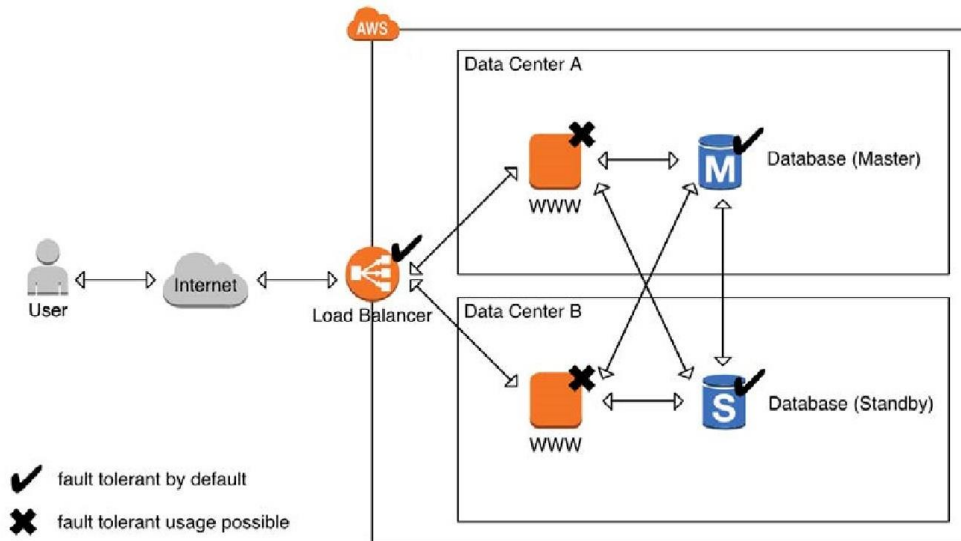


Figure 5 Building a fault tolerant system on AWS

You now have a broad idea what you can do with AWS. Generally speaking you can host any application on AWS. You will now learn how you interact with AWS to turn the system diagrams into reality.