AUGMENT ON-PREMISES SHAREPOINT WITH AMAZON WEB SERVICES

Use Cases for SharePoint on AWS Infrastructure as a Service

Presented by: Brian Laws, Managing Consultant, Summit 7 Systems

Date: 10/09/2014

SUMMIT 7 SYSTEMS, INC.
71 TOWN CENTER DRIVE
HUNTSVILLE, AL 35806
(256) 585-6868
WWW.SUMMIT7SYSTEMS.COM

TABLE OF CONTENTS

INTRODUCTION

SharePoint Server is one of the more powerful, flexible, and complex platforms from Microsoft. It has grown over the years to incorporate many different types of enterprise workloads, from document management to business intelligence to enterprise social. As it's done so, the amount of resources required to support the platform has grown, as have its operational complexities. At the same time, the cloud has continued to develop and has become a real, viable option to some and a necessity to others. SharePoint has, however, generally stayed on-premises (within the walls of a corporation's data center). At last, the maturation of cloud vendors like Amazon Web Services (AWS) has made it possible to consider cloudy alternatives.

As we will discuss below, there any number of challenges which may prohibit an organization from unlocking the full potential of SharePoint. This might be the physical resources available, the up-front cost of provisioning new resources, or the time it would take to do so. With some careful planning and engineering, an organization can use AWS to incrementally add capacity or capabilities, or it could introduce entirely new workloads that previously were out of reach. In this whitepaper, we will discuss at a high level some specific uses cases for how you can take advantage of the power of AWS' Infrastructure as a Service (IaaS) offering to augment your existing on-premises SharePoint practice.

THE CHALLENGES OF SHAREPOINT ON-PREMISES

It is no small task to host a SharePoint farm. There are many elements in SharePoint that can be (and in some cases actually are) significant enough to be platforms in their own right. Ironically, success in SharePoint can make managing it even harder, especially when it's time to grow the farm. Doing so while hosting it on your own servers in your own data center can be incredibly challenging. The capabilities of AWS, however, can offer an answer to many of these pains.

The following are some of the common challenges that an organization can face when managing and growing an on-premises SharePoint practice.

1. Server Capacity – There is a fixed capacity on premises, with a limited number of servers to provide services. Organizations might not provide enough CPU, memory, or disk, and adding more can be time-intensive, difficult, and expensive.
2. Disk Capacity – Enterprise-class storage arrays are expensive to purchase and maintain. As the amount of data grows, so must the size of the storage infrastructure to support it.
3. Supporting Infrastructure – Servers, storage, and networks require electricity, battery backup, cooling, floor space, racks, cabling, staff, etc. to install and maintain. These are limited resources and do run out.

4. Speed-to-Market – When it is time to provision a new farm or scale an existing one, if there is no remaining capacity in the data center, then it can take up to six months to purchase, order, receive, and install new hardware before SharePoint can even be installed or capacity added. By this point, the original business opportunity may have passed. The alternative is to over-purchase or over-provision, but this consumes funds that can be used elsewhere.
5. Expense / Costs – Physical infrastructure is an up-front Capital Expense (CapEx).
6. Charge Back – With so many elements comprising a SharePoint farm (server, storage, power, cooling, etc.), it can be difficult to quantify the true cost of a solution. This can make it difficult to charge a business unit for the cost of their solution.
7. Security of Mixed Workloads – It can be complicated and risky to host both external and internal SharePoint sites within the data center. Opening services to the Internet increases the attack surface so this needs to be carefully managed.
8. Aging Environments – Resources, both physical and soft, age and eventually need to be replaced or upgraded. This can be complicated and costly because doing so usually requires bringing additional resources online (especially since SharePoint 2013 does not support in-place upgrades). This, then, raises the challenges listed above.
9. Uncooperative IT – It is unfortunate, but for many business users, the prospect of requesting from IT a new SharePoint environment or additional resources is a painful, if not futile, prospect. Sometimes an IT organization can be uncooperative or even impede business agility.

THE CLOUD TO THE RESCUE!

With AWS IaaS, you can extend your data center and provision a near-limitless amount of resources. Be they hybrid or entirely standalone (unconnected to your corporate resources), you can provision however many servers and storage you need. With the many options available, you can choose VMs sized to fit whatever needs you have.

How can AWS meet the challenges outline above?

1. Server Capacity – AWS offers virtually limitless server capacity. Existing virtual machines (VMs) can be quickly scaled up, and additional VMs can be provisioned with a few clicks of the mouse or via automated procedures.
2. Disk Capacity – Nearly limitless, inexpensive storage of a variety of types is available immediately and without the need for additional management.
3. Supporting Infrastructure – With AWS, there is no infrastructure for you to be responsible for. It is included in the cost of the service.
4. Speed-to-Market – New servers, storage, networks, etc. can be provisioned with a few clicks and brought online in minutes (if not immediately). With automation, a server can be added to a SharePoint farm within an hour. This enables the business to be more agile and act quickly.

5. Expense / Costs – AWS represents an ongoing Operational Expense (OpEx).
6. Charge Back – As a billed service, it can be simple to determine the cost of hosting the solution. Depending on the design, the business unit could potentially even be charged directly.
7. Security of Mixed Workloads – In AWS, external SharePoint sites can be hosted separately from any internal resources, potentially eliminating the need for inbound connectivity from the Internet at all.
8. Aging Environments – New environments can be spun up in AWS quickly and with minimal financial commitment. This eases the testing and evaluation of upgrades as well as provides infrastructure into which a farm can be migrated, all without the need to purchase hardware. And as is the case with all IaaS, there is no concern for replacing physical hardware.
9. Uncooperative IT – In AWS, although it is not encouraged, business users have the capability to provision resources without the involvement of IT. They may, for example, directly engage a consultant to provision according to their needs.

HYBRID TOPOLOGIES

When deploying a cloud architecture, it is important to consider whether and how to extend your data center. The architectures easiest and cheapest to implement and maintain are entirely contained in AWS without connectivity to your corporate network. In this model, AWS serves as an entirely separate data center that is entirely isolated from your own. However, you can extend your data center into the cloud via a number of techniques, giving you a hybrid topology. The initial configuration of a hybrid topology can be complex, yet it can unlock significant benefits such as a unified authentication experience and easy access to and from corporate systems. Most of the use cases described below require a hybrid data center.

With AWS, there are two primary means to implement connectivity for a hybrid data center:

- Virtual Private Network (VPN)
- AWS Direct Connect

The VPN in AWS is like any other traditional VPN you might already be familiar with (see below, Figure 1 – Single VPN Connection). It securely connects your Amazon Virtual Private Cloud (VPC) network to your corporate network via a virtual private gateway in your VPC and a hardware or software VPN gateway located in your facility.

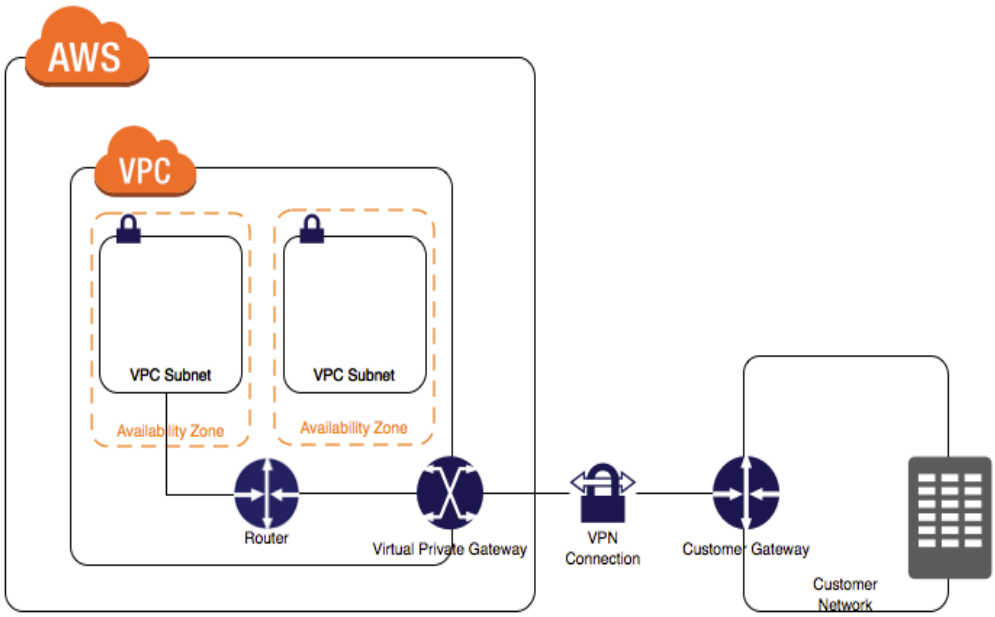

Figure 1 – Single VPN Connection

You may also connect your VPC(s) to multiple locations via multiple VPN connections (see below, Figure – 2 Multiple VPN Connections).

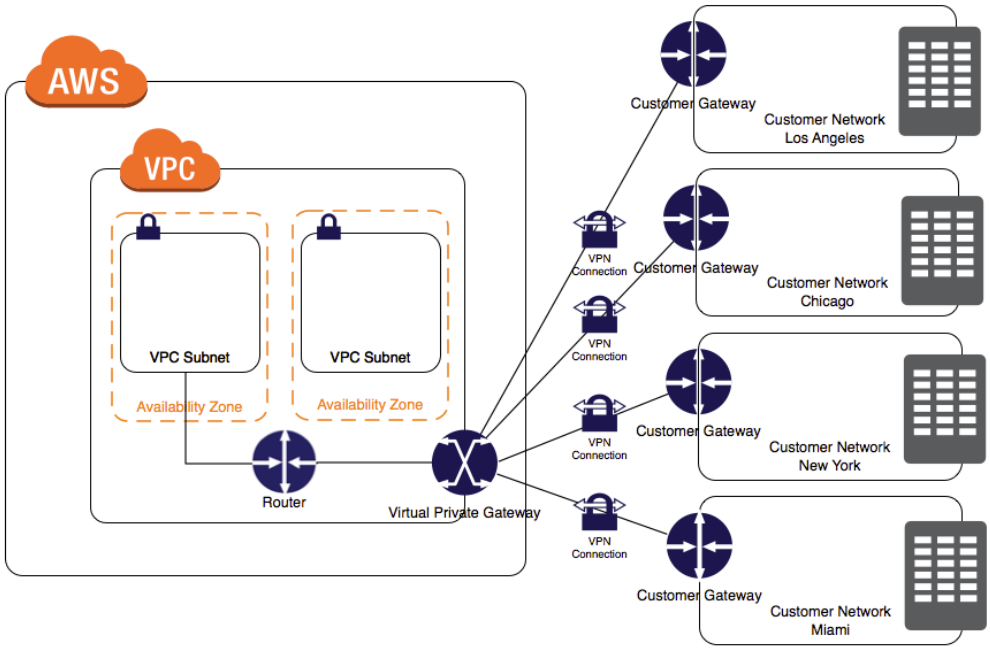

Figure 2 – Multiple VPN Connections

It can be relatively inexpensive to implement a VPN with AWS, costing (as of this writing) as little as $500 for the VPN device and $0.05 an hour ($36/month) per VPN connection. As with all VPNs, traffic travels across the Internet (secured via SSL encryption) and thus performance will be dictated by your bandwidth and inconstancies across the Internet. Available bandwidth and higher latencies are important to consider when designing a solution based on VPN.

AWS Direct Connect is an alternative means to connect your network to AWS. Direct Connect bypasses the Internet and your ISP altogether since it uses a private 1 gigabit or 10 gigabit fiber-optic cable. Not only does this provide a secure connection, it also significantly decreases latency. It can, however, be more expensive than VPN, and is limited geographically. Some of this cost might be offset by potentially allowing you to decrease your bandwidth with your ISP since the traffic to and from AWS is not traversing the Internet. Note that the latency, although improved over VPN, is still high enough to disqualify it for consideration in establishing a SharePoint stretched cluster (that is, a farm with servers in multiple data centers). Seriously consider implementing Direct Connect if your solution calls for the transfer of large datasets (such as in the Business Intelligence Farm use case). In any number of hybrid scenarios, AWS Direct Connect can make hybrid topologies much more seamless and secure.

With both VPN and AWS Direct Connect, it is important to consider redundancy in your design. What would be the impact if that connection went down? If budget allows, make sure to implement a second, redundant connection in order to provide for automatic failover. Additionally, consider implementing a second VPN (not just a VPN connection) in case your VPN device fails.

USE CASES FOR SHAREPOINT ON AWS

So far, we've seen how AWS can overcome many of the challenges organizations face in a traditional data center while hosting a SharePoint farm, and we've seen the options for connecting your corporate network to AWS. Now let's take a look at some specific use cases for using AWS to augment your SharePoint practice. What are some workloads that can be moved into the cloud or added altogether?

General Concepts

In each of the below use cases, the following concepts will be true.

1. It may not be necessary to provide inbound connectivity from the Internet. This would be the case if the rest of your virtual infrastructure is inside your VPC, or if your corporate network is connected via VPN or Direct Connect. In this instance, the components deployed in AWS will be entirely private.
2. If services in AWS should be accessible from the Internet, an Internet Gateway will be required to provide inbound connectivity to your VPC. As with a physical data center, it will be important to carefully architect your incoming connections to narrow your attack surface.
3. It is highly recommended that a minimum of one Domain Controller (serving the global catalog and DNS) be deployed in AWS. Not only does this provide for improved performance, it also provides resiliency in case the connection with the corporate network is lost. For more, please see "Active Directory Domain Services on the AWS Cloud: Quick Start Reference Deployment" (referenced below).

4. In non-hybrid scenarios or when administrative access is needed outside of the corporate network, it will be necessary to deploy a secured remote-access environment. This might take the form of jump boxes (remote-desktop into an Internet-exposed VM, then RDP to internal resources) or Remote Desktop Gateways. For more information, see "Deploy Remote Desktop Gateway on the AWS Cloud" (referenced below).
5. Consider your availability needs when deploying services in AWS. It is generally recommended to deploy all EC2 instances with high availability. With SharePoint, that means at least two web servers (behind an Elastic Load Balancer), at least two application servers, a two-node SQL AlwaysOn Availability Group cluster, two domain controllers, etc. The pairs should be placed in separate Availability Zones within the same AWS Region. The latencies between zones meet the Microsoft requirements of a stretched cluster. However, if your Recovery Time Objective (RTO) does not require that level of protection, then deploy only the number of instances the workload actually requires. For more information on how to deploy SharePoint in AWS with high availability, see "Microsoft SharePoint Server 2013 on the AWS Cloud: Quick Start Reference Deployment" (referenced below).

The below diagrams are intentionally simplified and do not demonstrate, for example, high availability across Availability Zones or scale-out architectures (like separate distributed cache farms).

Primary On-Premises Farm (in the Cloud)

Although it basically goes without saying, you can host your entire SharePoint environment entirely on AWS without any on-premises deployment at all. With this topology, AWS becomes your on-premises deployment, see Figure 3 – IaaS, Running an On-Premises Farm in AWS.

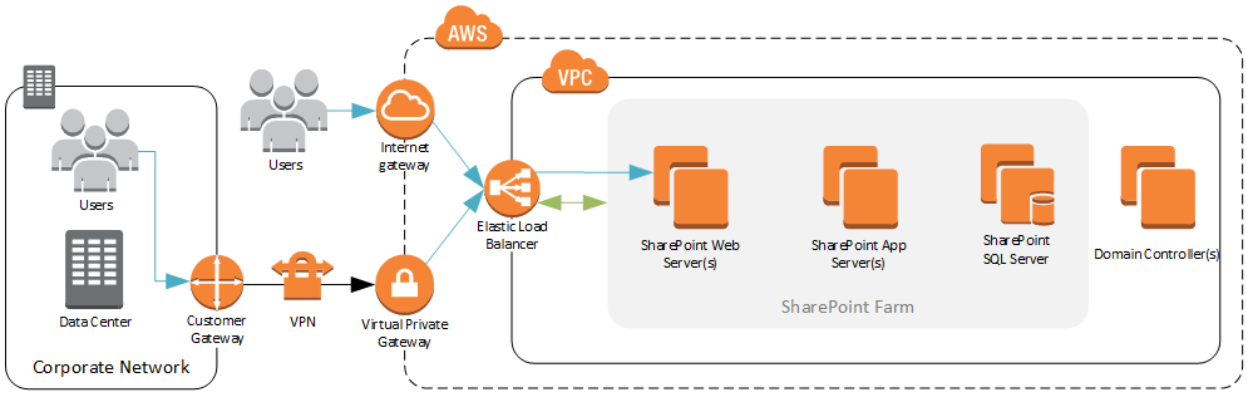

Figure 3 – IaaS, Running an On-Premises Farm in AWS

In this use case, since the primary farm is in AWS, it is not strictly necessary to have a hybrid topology with a connection back to the corporate network. This model is ideal for green-field deployments or companies without a data center.

Secondary Content Farm

As discussed above, there are significant challenges when the SharePoint farm needs to grow. It may take too long to provision new hardware, there might not be sufficient infrastructure, there might not be enough disk storage, etc. If you need to grow your on-premises farm but there isn't the capacity in the data center, then you can provision a secondary SharePoint farm to hold the additional content (see below, Figure 4 – Secondary SharePoint Farm in AWS). It is not possible to add AWS servers to the primary on-premises farm, since that would constitute a "stretched farm" (and is thus unsupported by Microsoft). The AWS farm would be a separate farm. However, it would subscribe to key service applications that have been published in the on-premises farm (for more, see "Share service applications across farms in SharePoint 2013," referenced below). User Profile, Managed Metadata, Secure Store, and Search are the key service applications to share, and doing so will provide for a fairly seamless experience for your users.

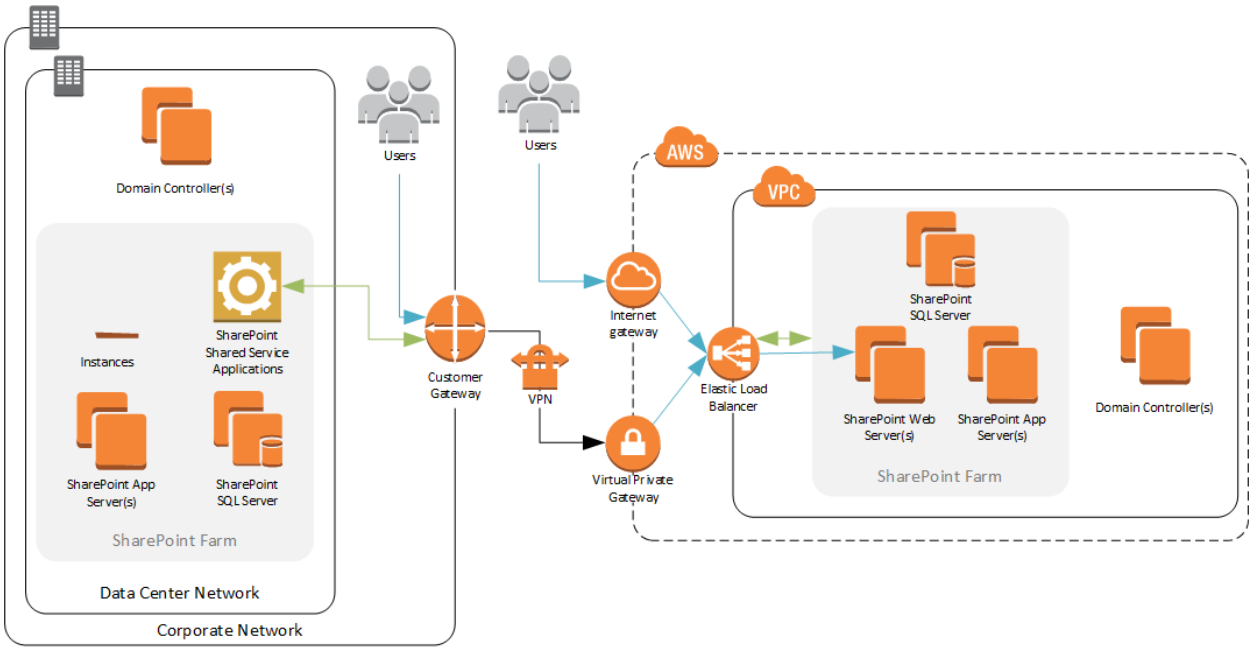

Figure 4 – Secondary SharePoint Farm in AWS

Once the service application federation has been set up, the farm can then be scaled as necessary to meet your needs.

Note that this architecture requires that each farm host a unique set of web applications or host-named site collections. Both farms will not be able to serve content for the same URL.

Extranet Farm

SharePoint is commonly used as an extranet in order to interact with customers, vendors, or the public at large. Unlike most traditional SharePoint deployments, an extranet is designed to host incoming Internet traffic. This is quite unnerving for most organizations since it introduces a significant security risk. Normally, a separate farm is deployed in a DMZ along with tightly controlled firewalls. This normally

requires separate hardware and a separate farm, and it significantly complicates the infrastructure landscape.

An alternative is to deploy an extranet farm entirely in AWS. For maximum security, the extranet farm can be 100 percent segregated from the corporate network. This would, however, require a dedicated domain controller with separate user accounts, and it would complicate SharePoint publishing models. An alternative would be to deploy a hybrid model like the Secondary Content Farm but then rely upon strict firewall/routing policies to protect the corporate network. Finally, it is possible as well to implement a traditional DMZ-based topology with the SharePoint farm in a separate isolated subnet in the VPC.

Search Farm

The hardware requirements in SharePoint 2013 are substantially greater than prior versions. This is largely due to the Search service (a direct descendant of FAST Search Server). A properly architected and sized Search service can take as many resources as a full SharePoint 2010 farm. As such, it can be difficult and expensive to provision a full Search farm on-premises (see below, Figure 5 – Search Service).

Additionally, Search crawl/enrichment/analytics can be a dynamic workload. Ongoing incremental crawls will have significantly less impact than a full crawl. In AWS, an excellent option would be to provision a Search service with extra instances hosting these components, except only power them on whenever there is a need for a full crawl. This can be done automatically with the AWS Auto Scaling service and this can significantly save on costs since the instances will be shut down for the large majority of the time. This strategy could also be used if content sources are crawled infrequently. For example, if you need to crawl aws.amazon.com once a month, power-on the already-configured instances before the crawl starts, and then shut them down after the crawl and the analytics processing completes.

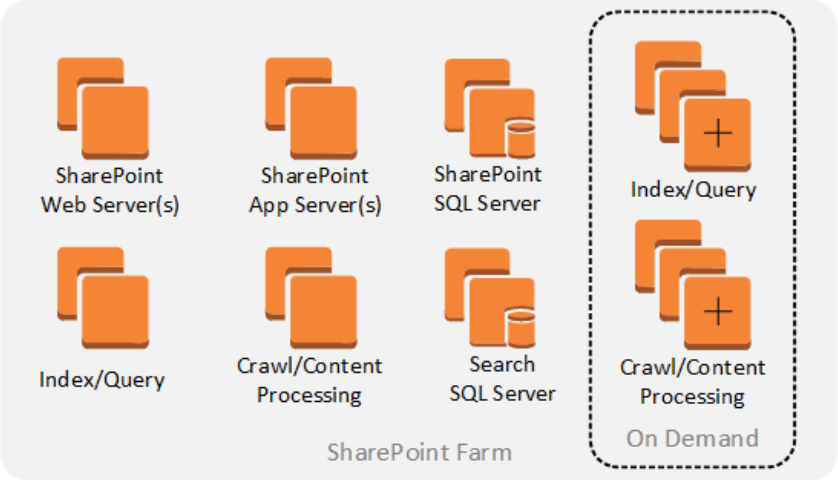

Figure 5 – Search Service

A Search index can grow very large, and it can consume a significant amount of resources. Early in a SharePoint project, it is hard to properly size a Search implementation since the size of the corpus is rarely understood. Plus, success in Search often results in the addition of more content sources. A small Search index can thus become quite large. With an AWS Search farm, it is easy to add additional capacity

as it becomes necessary. Simply provision a new farm server, update the Search topology to give the new server an Index component, and then let Search replicate the index to it. There is no need to over-provision at the beginning of a project and thus incur extra cost.

Disaster Recovery Farm

Being able to fully recover a SharePoint farm off-site to protect against a disaster is complicated (deserving its own whitepaper) and expensive. It requires a second physical location with its own infrastructure, hardware, staff, licenses, etc. The cost of all of this is even harder to bear when you realize it all is mostly unused throughout the year. It is such a challenge that many organizations do not even attempt a disaster recovery plan beyond backing up the farm or even just the SQL Server databases.

Instead, consider utilizing AWS as a virtual disaster recovery site (see below, Figure 6 – Disaster Recovery Using AWS). The technologies and processes used are similar to those deployed with a physical disaster recovery site. In "Choose a disaster recovery strategy for SharePoint 2013" (reference below), three primary recovery options are presented: Cold Standby, Warm Standby, and Hot Standby. With AWS, Hot Standby becomes much more of a viable option since only the minimum number of instances need to be running at any given time, with the full farm only being brought online when implementing the D/R plan (plus regular patch/update periods). This decreases the cost of a disaster recovery solution.

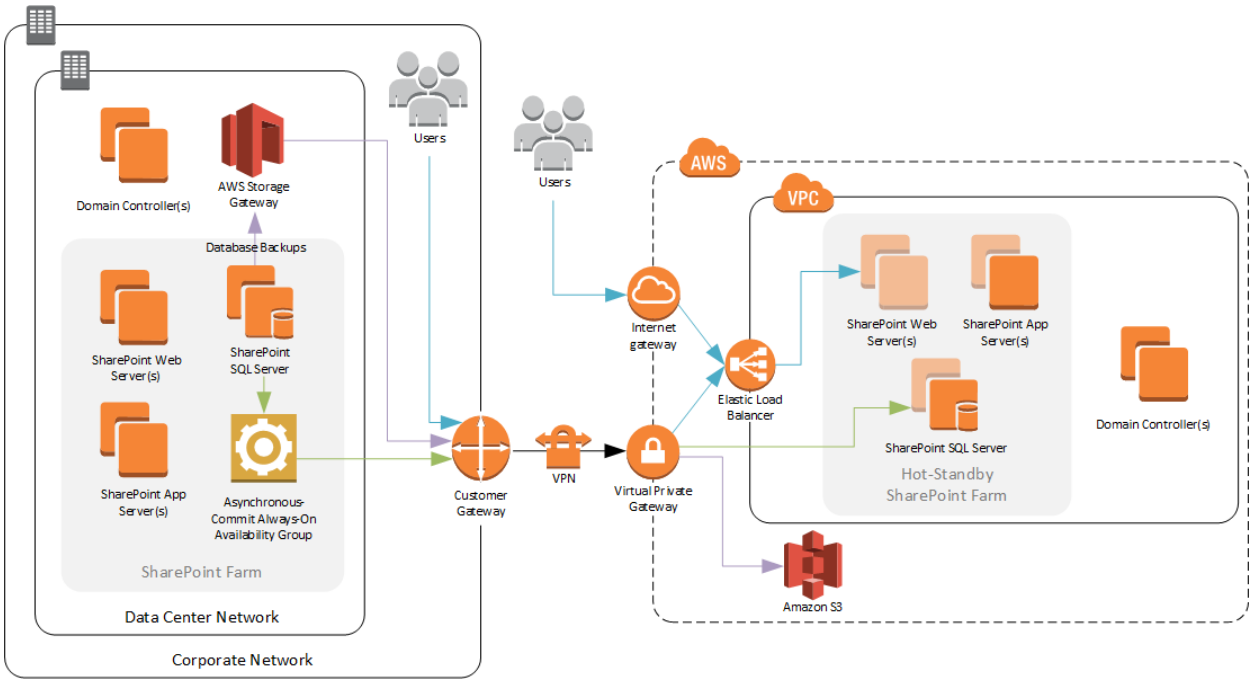

Figure 6 – Disaster Recovery Using AWS

The key to using AWS as a disaster recovery solution for SharePoint is to implement SQL Server AlwaysOn Availability Groups (AAGs). New with SQL Server 2012, AAGs replace Database Mirroring and will send transactions to a remote location asynchronously (or even synchronously if so desired). AAG

asynchronous commit is now supported for most SharePoint databases. See Spencer Harbar's article on AAG Async Replication support with SharePoint 2013 (referenced below).

With any of the disaster recovery scenarios, it is necessary to deploy a hybrid topology with VPN and/or AWS Direct Connect (highly recommended). Additionally, consider using AWS Storage Gateway to implement a storage gateway that serves as a target for SQL Server and farm backups. Doing so will automatically upload the backups to Amazon S3, thus protecting them from a site disaster.

Shared Services Farm

In the case of geographically dispersed organizations, a single, central SharePoint deployment can cause issues for remote locations. Any number of reasons, such as speed, latency, and bandwidth requirements, may prohibit this single, centralized farm. Since Microsoft does not support "stretch clusters," it becomes necessary to deploy a separate SharePoint farm at each location.

As discussed with the Secondary Content Farm use case above, service application federation can be leveraged to provide a unified experience with these local farms. It eliminates the duplication of data and configurations, and it decreases farm maintenance. To that end, a SharePoint farm can be deployed whose sole purpose it is to publish service applications for subscription by either local or remote SharePoint farms. These are known as shared services farms (see below, Figure 7 – Shared Services). A shared services farm can be implemented at any of the company's locations. Deploying a shared services farm in AWS, however, provides all of the other benefits discussed thus far, but it also eliminates a single physical location as a point of failure. If that facility were to experience an outage, then that failure would be felt at all remote locations as well. AWS will likely provider better availability. The front-end web role is not needed in shared services farms.

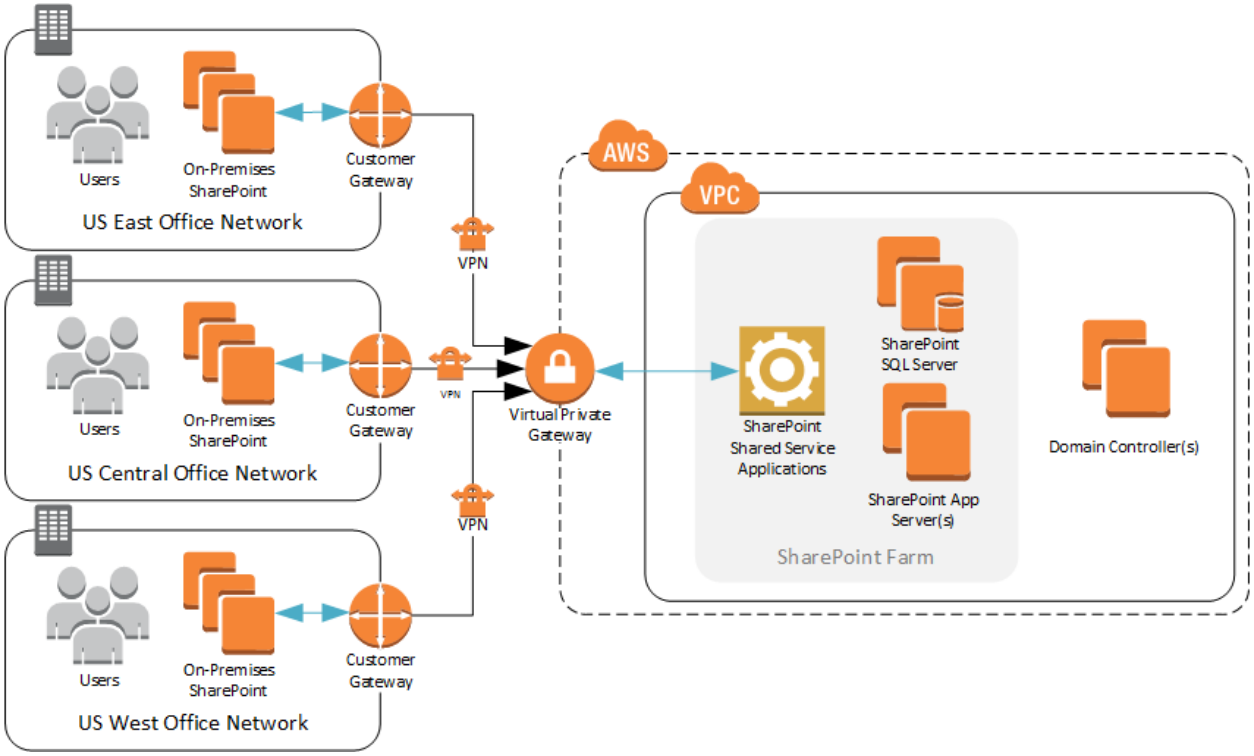

Figure 7 – Shared Services

Development Environment/End-User Sandbox

In organizations that do a lot of customization and development in SharePoint, or that do development/testing of third-party systems that in some way interact with SharePoint, developers often need a SharePoint environment of their own that they can work with. Additionally, it may be possible that end users require an environment that they can use as a sandbox or proof-of-concept. SharePoint farms, especially SharePoint 2013, require a significant amount of resources—usually more than their personal machines can provide. Development environments and sandboxes are ideal for the disposable computing that AWS provides (see below, Figure – 8 Development and End-user Sandbox).

The speed and ease at which development/sandbox environments can be provisioned and torn down is critical for this type of use case. Often, the need for such an environment comes and goes quite quickly. As such, it is important for there to be automation in place to make these farms quick and easy to deploy, each with consistent quality. CloudFormation can be used effectively to provision “stacks” on demand. A stack is a complete environment with everything it needs, from subnets to security groups to instances. After the user is finished with it, the stack can be removed as a unit as well. Additionally, any number of longer-term, shared farms could be stood up with or without isolation from each other to serve as a sandbox for users for testing, training, and proofs-of-concepts for third-party applications.

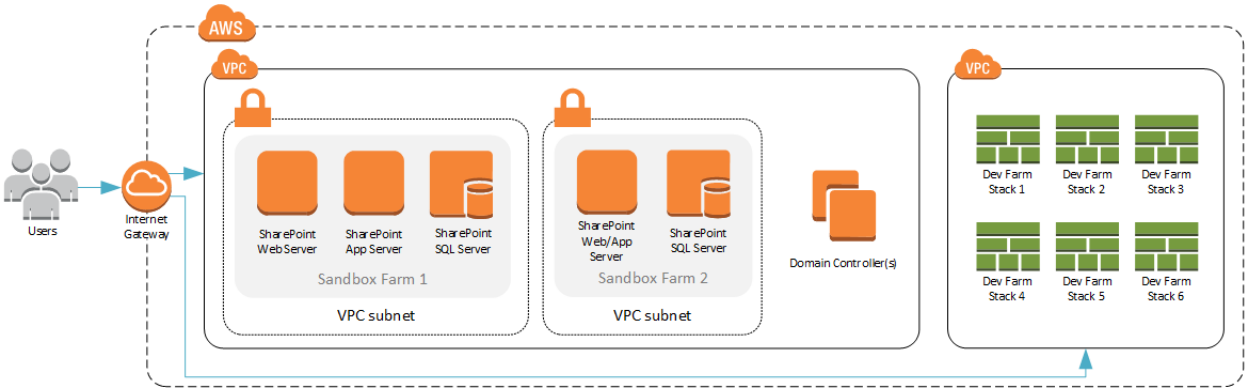

Figure 8 – Development and End-user Sandbox

Business Intelligence Farm

The final use case we will discuss is that of a Business Intelligence Farm in AWS (see below, Figure 9 – Business Intelligence Farm). Business Intelligence, or BI, is growing in importance in the corporate world as organizations realize the significant impact it can have focusing strategy and identifying opportunities or areas of concern. It's also growing in technical size and complexity. The amount of data to process is growing exponentially, and the technologies used to mine it for actionable information require more and more resources. It can be difficult to maintain this amount of physical resources in an on-premises data center. Additionally, traditional corporate procurement cycles are frequently unable to meet the rapidly changing demands of an organization. The pace of growth and innovation in the BI space can be too fast for traditional IT to keep up.

Because servers can be provisioned and grown on demand, and since there is nearly unlimited storage available, AWS is an ideal place to build a BI farm. It can be done with or without SharePoint. Although a SharePoint farm can be deployed with its powerful UI and analytical tools like PowerPivot for SharePoint, PerformancePoint, Reporting Services/Power View, and Excel Services, great value can be had in simply deploying into the cloud the data warehouse, analytics, storage, and reporting elements of the BI stack. SQL Server multidimensional and tabular data models and cubes, for example, can require significant amounts of memory and storage, resources that might be unavailable in the corporate data center. This isn't a problem for AWS, where you can provision on-demand instances like c3.8xlarge and r3.8xlarge, which provide 32 CPUs with 244 GB of memory.

The on-demand pricing model is another benefit of moving these workloads into AWS. Frequently, the cube- and report-building processes are run only periodically. They can be shut down when not needed, decreasing costs. Alternatively, roles can be greatly scaled out for periods of high need (such as quarterly or yearly reporting) and then scaled back down after the analysis is complete.

It is also possible to take advantage of the AWS database PaaS offerings. Amazon RDS provides managed relational database services, decreasing maintenance overhead and lowering the bar to entry to traditional database workloads. Amazon Redshift can be leveraged to provide a managed data warehouse

solution that can process petabytes of data. Elastic MapReduce, AWS' Big Data solution, can also be deployed. The on-premises equivalents of these technologies, especially Redshift and Elastic MapReduce, can be prohibitively expensive to build and maintain. This is not the case with AWS since there are no up-front costs.

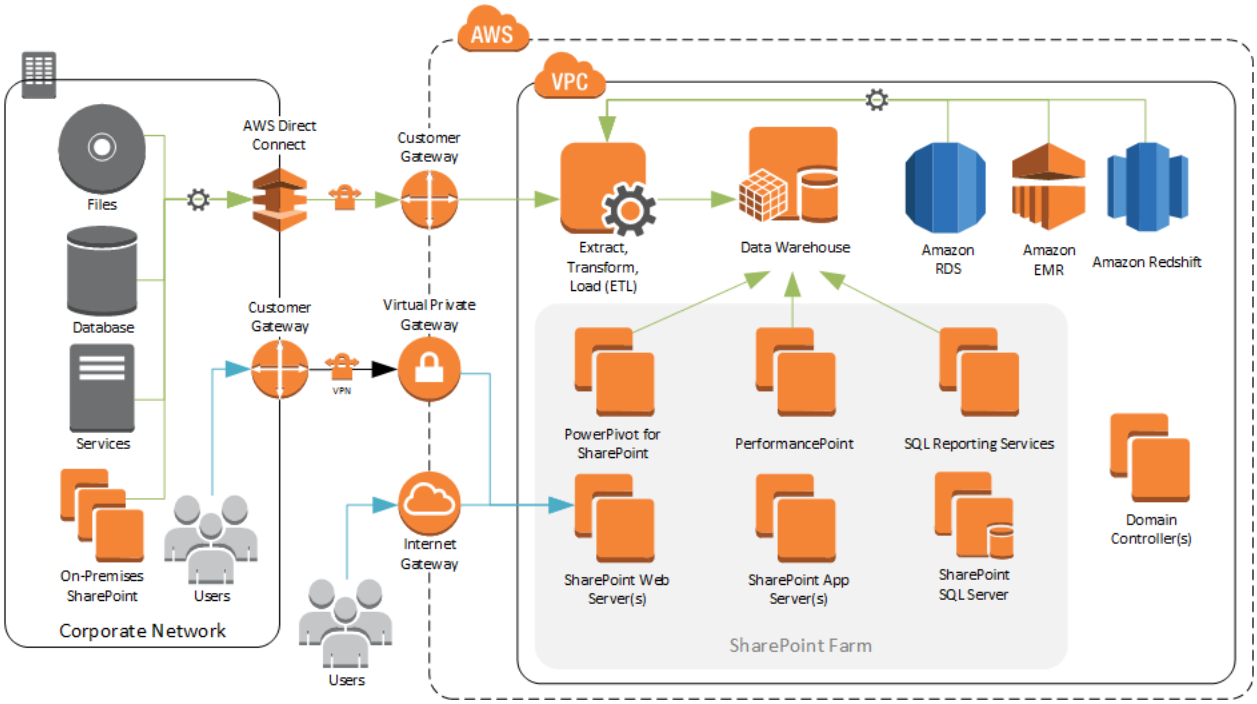

Figure 9 – Business Intelligence Farm

When deploying a BI solution in the cloud, it is highly recommended that the data be cached alongside the components in AWS (especially since the storage is so inexpensive). It is unlikely that connections from AWS-hosted services to on-premises data sources would be able to provide an efficient user experience for real-time analysis. Batch processing and storing the data in a data warehouse, in Amazon S3, or in Amazon DynamoDB would be ideal. Additionally, if the volume of data being pulled from the corporate data center will be high, consider investing in AWS Direct Connect for a high-speed, secure data channel. Not only will it decrease the amount of time needed to pull data, but it will also decrease your Internet consumption, giving more bandwidth to your users.

CONCLUSION

The power and capabilities of the cloud have grown tremendously over the past year or two. Providers like Amazon Web Services have enabled the hosting of true production workloads. Until recently, it would have been unthinkable to host a large SharePoint farm in the cloud. And yet with AWS, not only can we build production-ready SharePoint farms, we can do so in an agile and cost-effective manner. In this whitepaper, we discussed some ways you can use AWS to overcome some of the challenges you are likely

to find in a traditional on-premises data center. We also talked about several use cases in which AWS can be used to augment the services of an on-premises farm (especially when resources are scarce).

Even if your organization is unable or unwilling to move your SharePoint entirely into the cloud, it is still possible to expand your SharePoint practice by extending parts into AWS. Not only does it enable you to move existing workloads out of your data center (thus freeing resources for other use), but it also enables you to add *new* capabilities you otherwise wouldn't be able to on-premises. Hopefully, you see why hosting some (if not all) SharePoint in AWS can make a whole lot of sense.

REFERENCES

- AWS/Microsoft whitepapers:
  - Microsoft SharePoint Server 2013 on the AWS Cloud: Quick Start Reference Deployment: HTTP://AWS.AMAZON.COM/MICROSOFT/WHITEPAPERS/SHAREPOINT-2013
  - Active Directory Domain Services on the AWS Cloud: Quick Start Reference Deployment: HTTP://AWS.AMAZON.COM/MICROSOFT/WHITEPAPERS/AD-REFERENCE-ARCHITECTURE
  - Deploy Remote Desktop Gateway on the AWS Cloud: HTTP://AWS.AMAZON.COM/MICROSOFT/WHITEPAPERS/RDGATEWAY-REFERENCE-ARCHITECTURE
- Share service applications across farms in SharePoint 2013: HTTP://TECHNET.MICROSOFT.COM/EN-US/LIBRARY/FF621100(V=OFFICE.15).ASPX
- Choose a disaster recovery strategy for SharePoint 2013: HTTP://TECHNET.MICROSOFT.COM/EN-US/LIBRARY/FF628971(V=OFFICE.15).ASPX
- Support for SQL Server Always On Async Replication with SharePoint 2013: HTTP://WWW.HARBAR.NET/ARCHIVE/2014/03/20/SUPPORT-FOR-SQL-SERVER-ALWAYS-ON-ASYNC-REPLICATION-WITH-SHAREPOINT.ASPX