World Wide Technology, Inc.

Federated Application Centric Infrastructure (ACI) Fabrics for Dual Data Center Deployments

March 13, 2015

Abstract

To provide redundancy and disaster recovery, most organizations deploy multiple data centers. These multiple data centers are used in conjunction with a disaster recovery plan to provide business continuity. This paper illustrates how the Cisco Application Centric Infrastructure (ACI) can be implemented as two fabrics, with federated controllers implementing an autonomous infrastructure with replicated tenant configurations that will provide for disaster recovery.

World Wide Technology Advanced Technology Center

WWT designed, architected and implemented this disaster recovery ACI use case in its Advanced Technology Center (ATC).

The ATC represents a significant investment in technology infrastructure with hundreds of racks of networking, compute and storage products used to demonstrate and deploy integrated architectural solutions for WWT customers, partners and employees.

Powered by a multi-tenant private cloud infrastructure, the ATC is organized into four groups of labs for research and development, testing, training and integration. Each lab addresses different phases for the introduction, evolution and lifecycle of technology products.

The ATC ecosystem is defined by the combined experience of WWT Consulting Systems Engineers, IT Operations and Professional Services Engineers, along with the knowledge of peers from manufacturing partners and customers. This ecosystem of organizations provides thought leadership from a multi-disciplined technology perspective aligned by the common goal of integrating the right technology solutions to address and resolve real-world technical and business challenges.

For this use case, WWT architects, engineers and programmers used the ATC's Next-Generation Data Center environment.

Cisco Partnership

This white paper is an example of the partnership between WWT and Cisco Systems in developing solutions for the next generation in data centers. The Cisco Application Centric Infrastructure (ACI) is designed to manage a system of network switches and compute resources through a redundant, centralized Application Policy Infrastructure Controller (APIC). This system is managed to support specific application requirements, through the instantiation of a tenant. Tenants provide isolation of applications within an organization or between organizations.

Disaster Recovery Challenges

One operational challenge of any disaster recovery plan is implementing the processes and procedures to maintain consistency between the primary data center and the backup data center. All of the components in the data center — security policies, virtualization and compute resources, and storage — must be in a consistent state to support the migration of applications between data centers.

Cisco Application Centric Infrastructure Overview

The Cisco's Application Centric Infrastructure (ACI) fabric consists of two components: a central controller called the Application Policy Infrastructure Controller (APIC) along with the Nexus 9000 series switches in a spine and leaf architecture. The APIC implements network and security policy while the switches handle the delivery of packets.

The APIC abstracts the network infrastructure and provides a central policy engine. All interaction to the APIC is through the northbound APIs. These APIs can be used by programming languages, such as

Python, to extract policy from one fabric and copy the policy to a second fabric. This enables the network operator to configure and manage the primary or master fabric and programmatically synchronize the backup (disaster recovery) fabric.

The existing ACI/APIC architecture does not incorporate this programmatic synchronization. At WWT, we have developed software to demonstrate how two ACI fabrics can operate autonomously for external connectivity, while sharing the same application policy for applications running on either fabric.

Design Overview

This design demonstrates how two ACI fabrics can be implemented in separate data center environments to provide disaster recovery. The common and infrastructure fabric tenants have an autonomous IP addressing and routing configuration, while the application tenants are synchronized between controllers. Each APIC controller cluster is identified as the primary or secondary instance, and changes, additions or deletions to the application tenants, are replicated from the primary to the backup controller.

Following the initial deployment of the secondary fabric, no manual configuration is required to maintain currency of the secondary controller. Application tenant configurations are managed through a Python module developed by WWT that programmatically synchronizes the two fabrics.

In the event of a disaster declaration of the primary fabric, the role of the surviving data center can be designated as the primary controller and the network operator may add, delete or update tenants as part of normal operations.

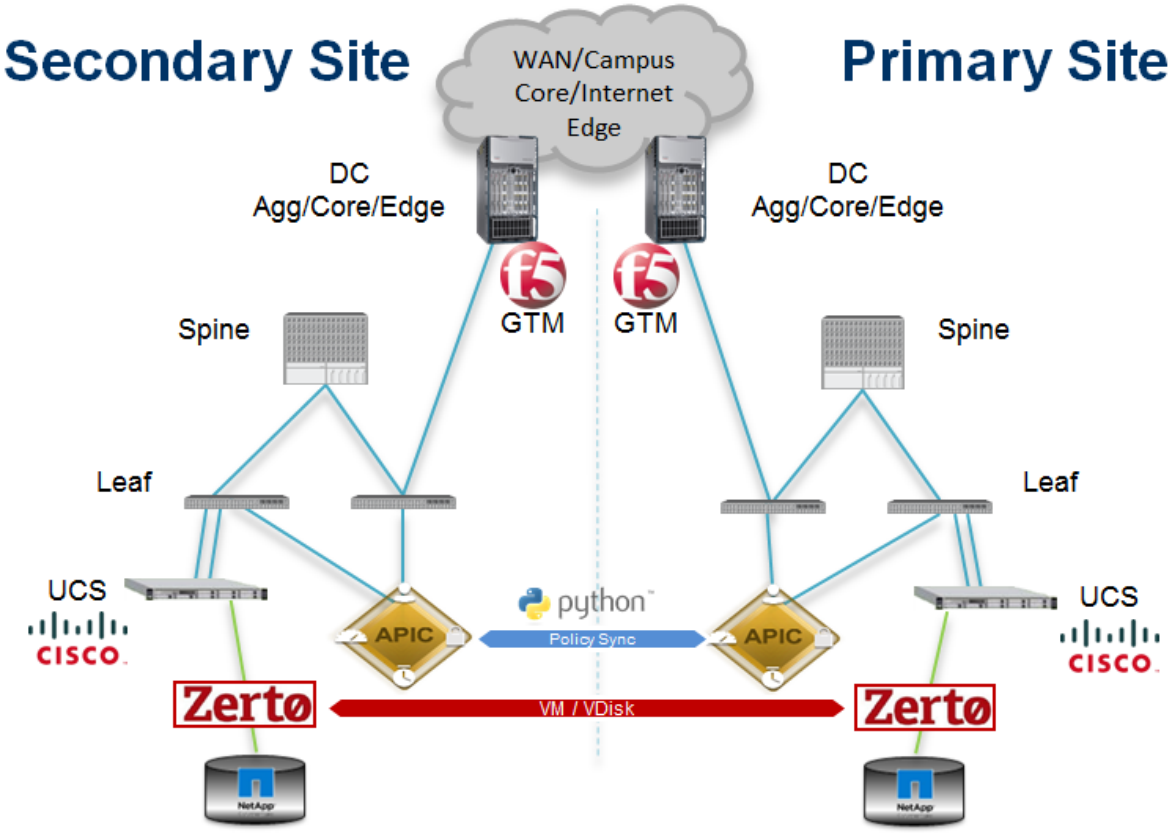
When the failed data center is restored, application tenant configurations are replicated to the recovered controller cluster. The roles of the recovered controller can again be designated as the primary if so desired.

Network Architecture

The network architecture is comprised of two independent data center fabrics with Layer 3 connectivity between them. Each data center has unique IP addressing namespace and connects to the WAN. A Data Center Interconnect (DCI) solution is not needed for fabric recovery.

The network architecture is described as an “East” data center and the “West” data center. In our operational model, the “East” data center is the primary data center and the “West” data center is the backup (disaster recovery) data center. This is illustrated in Figure 1 - Network Architecture.

Figure 1 - Network Architecture



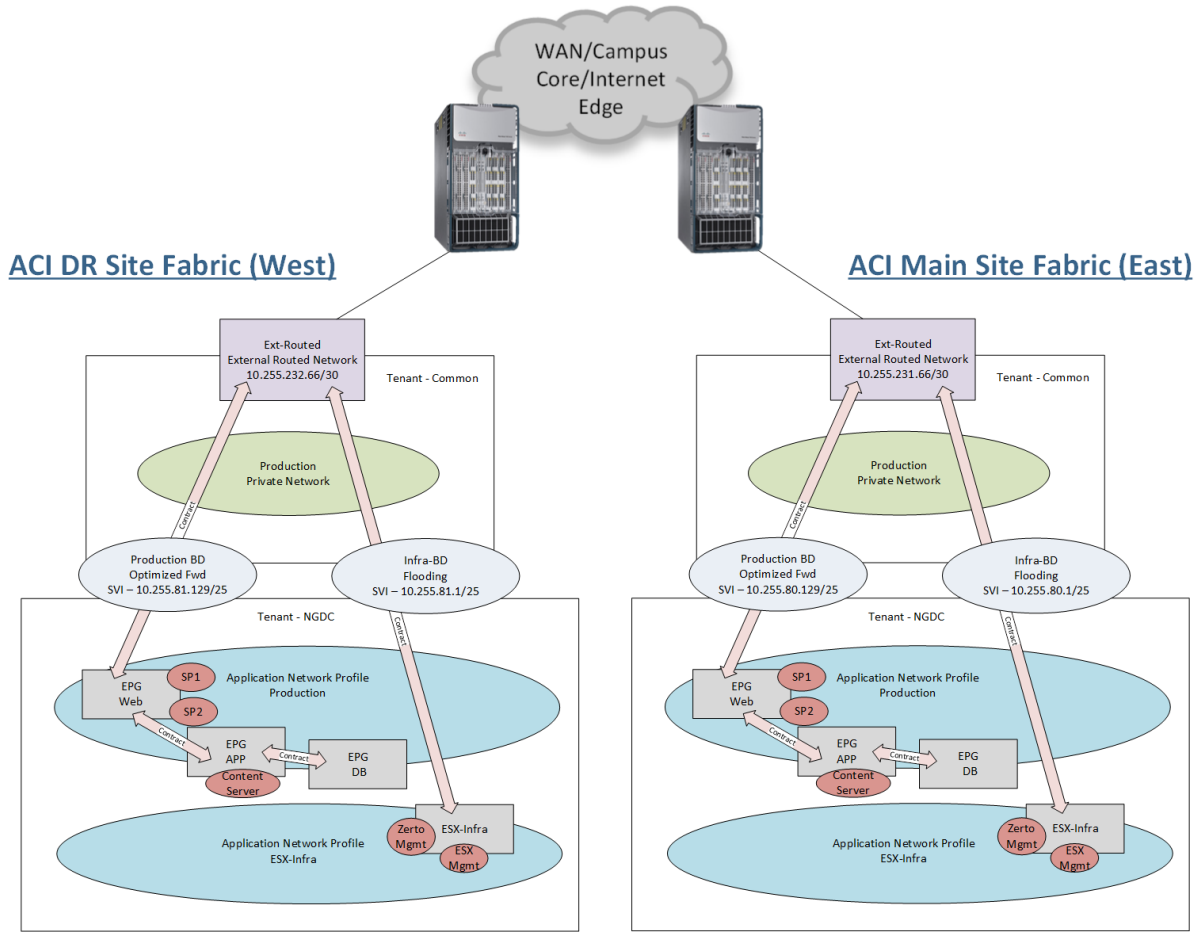
WAN Connectivity

External WAN connectivity for each data center is provided through the common tenant in the respective ACI fabrics. Each application tenant will access the WAN through the common tenant in the following manner. An Endpoint Group (EPG) will be created in the application tenant for connectivity purposes, (e.g., Web). This EPG will reference a bridge domain (e.g., Production BD) in the common tenant which has external connectivity. A contract will permit traffic to flow from the common tenant to the application tenant. Reference Figure 2 – ACI Logical View.

By using the common tenant for external connectivity, the network and security administrator can assign the appropriate network configuration policy, security contracts and policy, as well as firewall and load balancing services for the fabrics in each data center. The network policy is similar for each data center, but the IP addressing, and Bridge Domain and External Routed Network are specific to each site.

The Application (DevOps) teams will reference the common tenant configuration and configure application connectivity for intra- and inter-tenant communication through the Application Network Profile (ANP).

Figure 2 – ACI Logical View



Solution Components

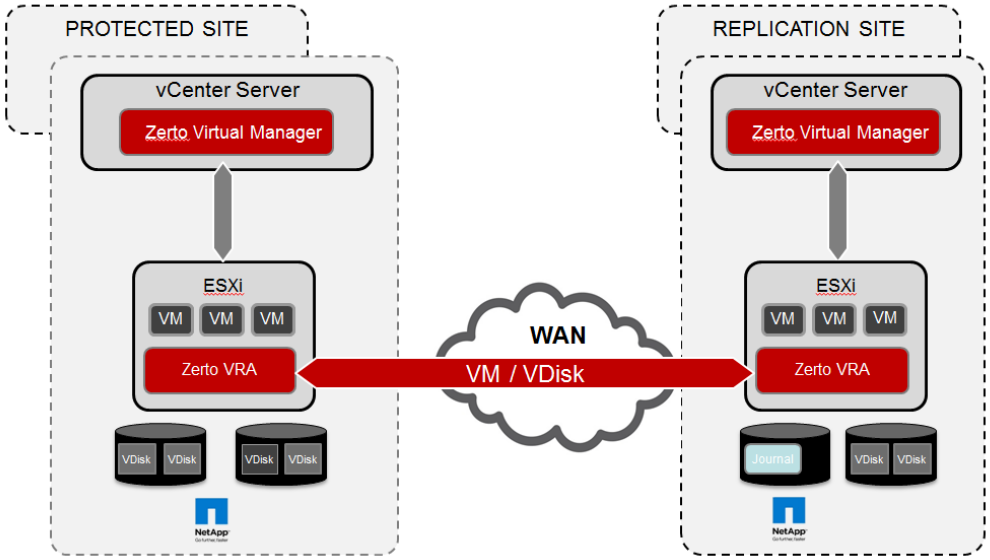
The design incorporates components typically found in data center environments. The role of the component and the specific product are shown in the following table.

Role	Product
Network Infrastructure Fabric	Cisco ACI - Nexus [2] 9396, [1] 9336 version 11.0(2m) [1] APIC version 1.0(2m)
Compute	Cisco UCS - C22 M3 Blade Servers
Storage Appliance	NetApp FAS 8000 - Network File System (NFS)
Storage Replication	Zerto Version 3.5 U4
Virtualization Hypervisor	VMWare ESXi and vCenter 5.5
DNS Global Load Balancing	F5 Global Traffic Manager (GTM) version 11.2

The application deployed to validate the solution is a two-tier Microsoft SharePoint 2010.

Zerto

Zerto is the first multi-tenant hypervisor-based data protection and automated recovery solution that supports both VMware and Hyper-V. Zerto is storage agnostic and replicates asynchronously, yet achieves recovery point objectives (RPO) usually measured in the seconds and recovery time objectives (RTO) measured in the minutes. Virtual Machine writes are replicated continuously from the main site to the DR site. Zerto's core capabilities include VM boot order automation, non-disruptive testing and IP address provisioning, all of which were used in this white paper design. Zerto is set up to replicate between two separate NetApp FAS 8000 Series Storage systems.



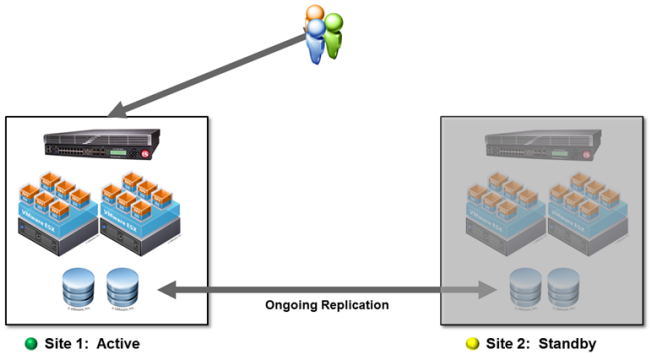
F5

F5 Global Traffic Manager (GTM) allows holistic management of multi-data center application delivery via intelligent DNS. GTM actively monitors application health at each data center and responds to DNS requests based on availability, performance and custom traffic engineering.

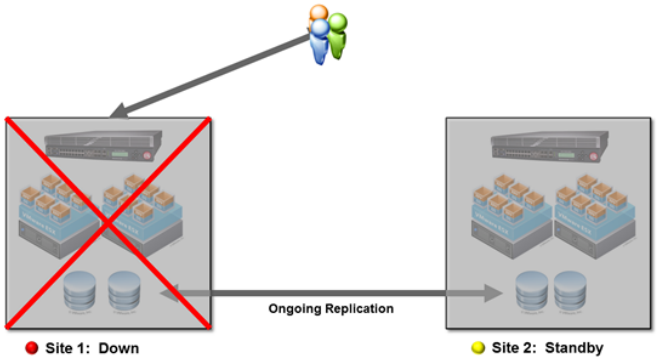
GTM uses a Wide-IP that maps a DNS entry to a pool of application instances spread across multiple data centers. In the event of a failure of a data center, the application instance in that data center will be unavailable and no longer be provided to DNS queries. Custom traffic engineering can be implemented to direct traffic based on variables such as application performance metrics, geo-location, round robin, etc.

The implemented design will have GTM contain application instances for SharePoint in each data center. Using Zerto within this design, SharePoint will only be active and available in a single data center at a time. The IP address for the active data center SharePoint instance will be given out to all DNS requests while still available.

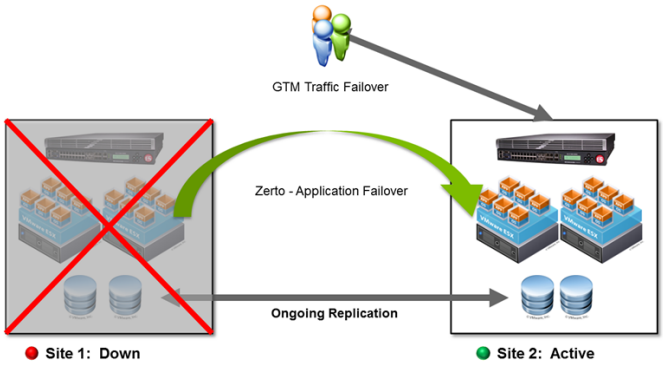
Normal Operations



During the Failover



After the Failover



Orchestration Software

The orchestration software was developed by WWT to programmatically synchronize the application tenant configurations from the primary data center to a secondary data center ACI fabric. The network administrator controls the designation of the role of each data center through a configuration file. We have named the program WWT APIC Federation Utility because the ACI fabrics operate autonomously from an external network connectivity standpoint, but are federated in that they share tenant configurations and can assume the role of primary and backup datacenter. The WWT APIC Federation Utility is written in Python and is initiated under Linux as a cron job. The recommended execution interval is every one to five minutes. The elapsed time to complete the WWT APIC Federation Utility execution depends on the number of tenants configured, however small deployments complete in less than five seconds.

The WWT APIC Federation Utility attempts to log in and establish a session with each APIC using the REST API northbound interface. If session establishment fails to either controller, the failed attempt is logged and no further action is taken. This state is representative of an active disaster.

If both controllers are online, the WWT APIC Federation Utility uses the REST API northbound interface to query a list of configured tenants and their respective modification timestamps (modTs), excluding the ACI fabric tenants infra, common and mgmt.

The WWT APIC Federation Utility then compares the list of tenants on the two controllers and takes these actions:

- If the tenant exists on the secondary, but not the primary, delete the secondary tenant.
- If the tenant exists on the primary, but not the secondary, copy the tenant configuration to the secondary.
- If the tenant exists on both primary and secondary, verify the tenant has not been manually modified since the last execution. Issue warning and take no action if the condition exists.
- If the secondary configuration has not been modified, delete the secondary tenant and copy the tenant from the primary controller to the secondary controller.

The WWT APIC Federation Utility generates a log file and, through the configuration file, provides an option to enable verbose debugging information to the console.

This logic accomplishes the goal of consistency between the application tenant policies between two data centers, without manual configuration of the backup site.

Caveats

During the replication process, the tenants are deleted and re-added excluding the common, infra, and mgmt tenants. There is a re-convergence time in the fabric in which policy is pushed and re-instantiated on the physical Nexus 9000 series switches. At that time, the EPGs associated with the tenant are not reachable. The observed time was between 20-35 seconds.

Conclusion

The Cisco Application Centric Infrastructure (ACI) is an innovative architecture where applications treat the data center as a dynamic, shared resource pool. This pool of resources is managed through a central controller exposing all configuration and management components through a northbound REST API. WWT is providing value to our customers by exploiting the programmatic interface of ACI to incorporate the fabric into a dual data center deployment.