Cisco's Massively Scalable Data Center

Framework

Network Fabric for Warehouse Scale Computer

Warehouse Scale Computer

The great democratization wave of computing begins with the migration of applications to the cloud-based, warehouse-scale computers. These computers are not confined to the chassis or racks or even to a single data center, but are collections of hundreds of thousands of computing components and petabytes of storage capacity that are deployed in a network fabric for delivering a few closely related web services to millions of internet users. The most distinguishing part of these computing systems is the massive scale of their software infrastructure, data repositories, and bare bones hardware platform. The scalability of these computing systems is significantly enabled by the network fabric, which has to meet a set of requirements that to date are only found in supercomputing environments.

Cisco's Massively Scalable Data Center (MSDC) is a design framework that focuses on key issues faced by architects and practitioners in this new design space where the design center on compute, storage, and network has shifted towards the far right on a scalability scale. This framework helps the designers to arrive at an approach for building a custom network fabric for their warehouse-scale computer..

Drivers of Cisco MSDC Design Framework

Cost Efficiency

A key driver in arriving at Cisco MSDC framework is to judiciously pick the network architecture to reduce the overall cost of the network without sacrificing fabric reliability. It is very simplistic to model the cost of a network as dollars per server access port. This metric is misleading because it does not include the cost of the fabric required to deliver symmetric bi-sectional bandwidth to each of these ports at scales exceeding several hundred thousand. Nor does it include the operational cost of troubleshooting and managing a large-scale network. In other words, any cost efficiency driven into the network architecture needs to target the fabric that weaves the access ports into a single fabric through connections. It turns out that the biggest wastage in the fabric is lower utilization of available and deployed network resources caused by multiple pathologies, including lack of multiple paths, congestion, contention, and suboptimal management infrastructure. Cisco's MSDC framework addresses this key driver by addressing the decisions that go into making a judicious trade-off between cost and increased utilization of existing investments.

Open Source Software Infrastructure

Another key driver for Cisco MSDC framework is the assumption that open source infrastructure and tool chains are the first choice in meeting the challenge of delivering applications to millions of users and managing hundreds of thousands of server units that are deployed to run these workloads. Open source is preferred primarily because of its maturity of the software infrastructure, monitoring and notification tools, and mechanisms to easily customize the software. Cost is no longer the main driver for using open source tools. Cisco's MSDC framework discusses the requirements posed by workloads generated by horizontally distributed applications and the tool chains that are deployed to manage, monitor, and troubleshoot these applications.

Bare Minimum Server Platform

Cisco's MSDC framework assumes that MSDCs will prefer a customized hardware platform, which could be a stripped down version of a traditional server optimized for an application. Having a homogeneous pool of servers under a centralized management helps in management and monitoring of large infrastructure.

Virtualization

Virtualization is not currently widely prevalent in MSDCs, but a growing interest exists in deploying virtualization for workloads that need to be tenant aware.

Data Center Power Usage

Data centers are increasingly using power usage effectiveness (PUE) as the metric to benchmark power efficiency and usage inside their data center. The ideal target for a data center PUE is 1.0, but that is a very difficult target to achieve. Cisco's MSDC framework discusses the major causes for increases in PUE and how Cisco's unified network and compute platforms help optimize the data center PUE.

Key Elements of Framework

Cisco's MSDC framework describes an abstract design for a MSDC class data center that an architect can use to design a customized network architecture. Having a framework influences the design choices that an architect makes as he/she designs the network fabric. Cisco developed a reference architecture based on this framework that is described in the Cisco Massively Scalable Data Center Overview. Key Elements of the framework described below serve as guideposts, determine research methodology, drive the plan for telemetry, and search for statistical relationships and mathematical correlations between various elements.

Topology

MSDC data centers are characterized by massively parallel workloads that require a rate of internode communication levels higher than what is supported by the existing network. This creates a network bottleneck. In some of these applications, like map-reduce, the internode communication needs to complete before the local computation can begin, while in others, like a query application, a simultaneous communication consisting of short messages exists between a node and every other node in the computation. This communication pattern is only increasing in use due to the use of SOA technologies for construction of web pages where a single web page can consists of hundreds of services.

The primary goal when building a network fabric for these applications is to provide enough bi-section bandwidth that communication-intensive parallel applications can maintain high levels of compute utilization by essentially reducing the wait or idle time of the CPU complex. An economic constraint to the topology design also exists. The cost of the network should not increase non-linearly with the size of the compute complex. Given these two constraints, it is easy to rule out any topology that has a single root. For example, where the topology is a single path routing tree of interconnected switches, as this topology is bottlenecked by the bandwidth that is available at the root. Cisco MSDC data center framework uses a fat-tree topology (k-ary). In general, for a k-ary three-stage fat-tree, the maximum number of nodes that can be connected is k^3/4. For example, an architecture based on this framework with k=48 can support approximately 27K hosts. Fat trees are also rearrangeably non-blocking, which essentially means that a set of paths exist between the two hosts even after the primary path is down. This multipathing can be increased for a given number of switching by increasing the height of the network topology.

Platform Selection

Using identical small state switches is cost effective up to a certain point. After that, the cost of the repeated packaging, PSUs, and cables begins to undermine the expected cost savings. In these environments, using high radix switches is proven to lower the overall cost for the network. The selection criteria for the spine is different from the leafs. In a folded Clos model where a leaf switch has "N" ports, the contribution to the server host count is N/2 for a non-blocking fabric. The remaining N/2 ports are consumed to build the fabric infrastructure. On a spine switch, all the ports are used for building the infrastructure. In other words, having a high radix (portcount) switch at the spine scales the infrastructure in a folded Clos architecture more than a low radix switch at the spine. In addition, because the high radix switch itself has a built-in switch fabric, it requires fewer layers of Clos that consume external-facing ports. For example, one could build a theoretical 7 or 9 stages Clos with only two layers involving a high radix switch with built-in switch fabric and a low radix switch with no switch fabric. Cisco's MSDC reference architecture is built using a Nexus 70XX series high radix switch and a Nexus 30XX series low radix switch. In our investigation, this is the optimal model for building multipath networks. It affords an architect the scalability of a Clos fabric without the cost inefficiencies incurred when using identical low radix switches in the network architecture.

Protocol Scalability

The selection of a set of protocols that enable the infrastructure and the routing of the flows through the infrastructure is perhaps the most important decision that a network architect makes at the design stage. The set of protocols have to serve two main purposes. The first is to enable self-configuration of the network fabric infrastructure itself. This involves self-discovery of the location of a device inside a multi-rooted topology like a Clos or fat tree. The second purpose is to discover and announce all the paths between the hosts connected to this network and judiciously choose the optimal link among the multiple available links. This load balancing among all the available paths is the main reason for data centers to migrate to the multipath network architecture.

Another factor that goes into selection of the set of protocols is the scalability of the set as a whole, or "the control plane" of the network fabric. The control plane should afford the Layer 2 plug-and-play architecture, but should scale like a Layer 3 network. It should also deliberately avoid using large flooding domains, but not limit host mobility from an any-to-any point in the network. In addition, consideration should be given to the state-holding capacity of a network switch. Using small state switches are more cost effective, but that requires a control plane architecture that summarizes multiple host addresses behind a single top-of-the-rack switch. Recent studies in academia and industry have pointed to a network architecture that espouses a Layer 3 control plane protocol organization to deliver scale-out networking. The primary reason cited for not using Layer 2 is that the use of the flat addressing requires every device in the network to learn the address of every host, causing state on the device to exceed the capacity of the device to hold state. Increasing the capacity to hold state on a device becomes uneconomical unless the ports on the device are also increased. In other words, for topologies that espouse low port count, small state devices in a network architecture, use of Layer 2 control plane is not cost effective. One downside of using Layer 3 control plane is the extra care one has to take to migrate virtual hosts across subnet boundaries. As part of the application state, the IP address and TCP connection information is cached at the host. Upon migration, such information becomes stale and needs to be refreshed without perceivable degradation at the user level. Layer 2 networks do not face this limitation because they do not rely on Layer 3 IP address of the physical machine.

Even selecting the set of Layer 3 protocols requires careful consideration of the amount of CPU utilization on the network device. Running two or more protocols simultaneously provisions the CPU of the network device for routing and away from other equally important tasks such as monitoring, alerting higher layer services, or just analytics in support of the traffic engineering scheme in place. The Cisco MSDC reference architecture uses a combination of BGP and OSPF for the control plane to discover and forward traffic among the hosts connected to the network.

Application Optimization

Modern horizontally-scaled applications do not rely on network middleware to abstract the network. Instead, they directly communicate with the network over a socket and account for multiple and simultaneous network failures. The drawback of this extra intelligence at the application layer is that it is not current to the underlying network architecture. In fact, very little or no topology or path discovery is done inside an application before it initiates a TCP connection. This lack of shared communication can be overcome by enabling programmability of the network, thereby enabling a developer to query the network controller for available paths and the characteristics of the available path.

Network Services

A number of services have migrated to the network from the host over the past decade. The most prominent of them are load balancing, IPAM, DNS, and Firewall. Incorporation of these network services into the core infrastructure is a challenge for multipath, multirooted networks as most of these services model the network as a tree with single root. Take, for example, load balancing where insertion of the load balancer requires in all but one case that the request and response traverse through the load balancer. In this network services is attached to the leaf as any other host, it will create a bottleneck for the flows through this network. A remedy for this pathology would be to incorporate the services in the top-the-rack switches, but again that would require state management for a cascading request. A cascading request is one where an innocuous looking single web request results in tens of internal requests to generate the correct response. In addition to the issue of service insertion, the issue of policy-basedrouting of the flow

exists. A policy may dictate multiple traversals through a firewall and this policy might change depending upon the flow. All of these complications make the network services design over a multipath network among the most challenging endeavors in network fabric design.

Managing, Monitoring and Diagnostics

Managing a network of this size has challenges that span the full spectrum of the network lifecycle starting from configuration installation and validation to routine maintenance. Open source tools exist that scale to enable monitoring, alerting, and reporting on a MSDC scale network. However, they require customization. Cisco MSDC reference architecture has customized plugins for popular open source toolsets like Nagios and Ganglia.

One of the surprising challenges in managing a large network is the cost and complexity of the cabling of the multitude of devices. The cost of cabling and implementing the new cabling is cited as one of the reasons for oversubscribing the network. Having a platform that can detect misconfigured cables can add to the cost savings and lower the total cost of ownership (TCO) of a network architecture..

Automation and Open APIs

At multiple points in the design process discussion so far, we have seen the need for dynamic configuration changes in response to changes in traffic patterns, policy mandate, or simply preference. These changes are manageable when the network is small; however, for MSDC class networks, this is a showstopper. Manual intervention or even passive supervision creates cost disadvantages for the service provider. For this reason, deploying programmable devices in a multipath network is a prerequisite for MSDC class network fabric. However, just having device programmability is not enough. Application innovation is creating requirements for programmability of the whole network. SDN is a set of technologies that aspires to operate the network as a distributed system with a middleware stack and controller..

Software Defined Network

SDN is a disruptive innovation that is challenging existing network architectures and products. In addition, SDN challenges existing service provider, IT organization, and network equipment supplier business models. This disruption is occurring first in the Massively Scalable Data Center. It is envisioned that this disruption will spread to other markets as end users, service providers, and IT organizations learn the benefits of SDN control. Cisco MSDC framework accounts for ubiquitous SDN in MSDC class data centers. SDN provides opportunities for network equipment vendors to expose product capabilities and leverage intellectual property in new ways. This creates new value for end users, service providers, IT organizations, and equipment vendors. The Cisco Software-Defined Network (CSDN) embraces and extends industry software-defined networking efforts to deliver flexible Cisco-powered networks supporting new architectures and application needs.

Conclusion

To deliver applications to millions of internet users, data centers are being built at massive scale in numbers and distribution. This is placing unprecedented demand on the network that transports the data between elements in the data center. Cisco MSDC design framework exists to meet the growing demand across the world for methodologies, best practices, and tools to meet this scalability challenge. This framework is primarily targeted towards a large data center; however, its concepts can help any data center that would like to scale up the data center while simultaneously leveraging existing investment.