Big Cloud Fabric™ is the next-generation data center switching fabric delivering operational velocity, network automation and visibility for software-defined data centers and cloud-native applications, while staying within flat IT budgets. Using hyperscale-inspired networking principles, software controls, and leaf/spine CLOS fabric delivered on open networking hardware, Big Cloud Fabric leverages software-defined networking (SDN) to make networks intelligent, agile, and flexible.

**BIG CLOUD FABRIC OVERVIEW**

Big Cloud Fabric (BCF) is the industry’s first data center fabric built using SDN controller software and open networking (white-box or brite-box) hardware switches.

Embracing hyper-scale data center design principles, the BCF solution delivers:
1. Intelligence by simplifying network operations and providing fabric-wide telemetry
2. Agility by enabling network automation to rapidly deploy application and services
3. Deployment flexibility powered by a scale-out architecture and open network hardware

Applications can now take advantage of high east-west bisectional bandwidth, secure multi-tenancy, and workload elasticity natively provided by BCF. Customers benefit from unprecedented application agility due to automation, massive operational simplification due to SDN, and dramatic cost reduction due to hardware (HW)/software (SW) disaggregation.

BCF supports both physical and virtual (multi-hypervisor) workloads and choice of orchestration software. It provides L2 switching, L3 routing, and L4-7 service insertion and chaining while ensuring high bisectional bandwidth. The scalable fabric is fully resilient with no single point of failure and supports headless mode operations.

**ARCHITECTURE:**

Software Defined Networking (SDN) fabric architecture refers to a separation of the network’s data and control plane, followed by a centralization of the control plane functionality. In practice, it implies that the network’s policy plane, management plane, and much of the control plane are externalized from the hardware device itself using an SDN controller, with few on-device off-load functions for scale and resiliency. The network state is centralized but hierarchically implemented, instead of being fully distributed on a box-by-box basis across access and aggregation switches.

Controller-based designs not only bring agility via centralized programmability and automation, but they also streamline fabric designs (e.g. leaf-spine L2/L3 Clos) that are otherwise cumbersome to implement and fragile to operate in a box-by-box design.

---

1 See specific hypervisor and orchestration support in a later section of this datasheet.
The BCF architecture consists of a physical switching fabric, which is based on a leaf-spine Clos architecture. Optionally, the fabric architecture can be extended to virtual switches residing in the hypervisor. Leaf and spine switches running Switch Light™ Operating System form the individual nodes of this physical fabric. Switch Light Virtual running within the hypervisor extends the fabric to the virtual switches. Intelligence in the fabric is hierarchically placed—most of it in the BCF Controller (where configuration, automation and troubleshooting occur), and some of it off-loaded to Switch Light for resiliency and scale-out.

**BIG CLOUD FABRIC SYSTEM COMPONENTS**

- **Big Cloud Fabric Controller** — a centralized and hierarchically implemented SDN controller available as an HA pair of hardware appliances for high availability.

- **Open Networking Leaf and Spine Switch Hardware** — the term ‘open networking’ (whitebox or britebox) refers to the fact that the Ethernet switches are shipped without an embedded networking OS. The merchant silicon networking ASICs used in these switches are the same as used by most incumbent switch vendors and have been widely deployed in production in hyperscale data center networks. These bare metal switches ship with Open Network Install Environment (ONIE) for automatic and vendor-agnostic installation of third-party network OS. A variety of switch HW configurations and vendors are available on the Big Switch hardware compatibility list.

- **Switch Light Operating System** — a light-weight open networking switch OS purpose built for SDN

- **Switch Light VX (optional)** — high-performance user space software agent for KVM-based Open vSwitch (OVS)

- **OpenStack Plugin (optional)** — BSN Neutron plugin or ML2 Driver Mechanism for integration with various distributions of OpenStack

- **VMware vCenter Extension / GUI Plugin (optional)** — built-in network automation and VM Admin visibility for vSphere server virtualization; NSX network virtualization; vSAN storage virtualization, and VMware integrated OpenStack (VIO)

- **Container Plugin (optional)** — BCF plugin for various container orchestrators for container-level network automation and visibility

**DEPLOYMENT SOLUTIONS**

BCF is designed from the ground up to satisfy the requirements of physical, virtual or combination of physical and virtual workloads. It supports a wide variety of data center use cases, including:

- VMware SDDC workloads (vSphere, NSX, Virtual SAN and VIO)
- OpenStack including NFV
- Containerized workloads
- Private clouds
- Virtual desktop infrastructure (VDI) workloads
- Big Data / High Performance Computing
- Software Defined Storage (SDS)
The BCF fabric can be designed to support the above listed deployment scenarios using a combination of open networking Ethernet switch options. A few examples are listed in the table shown in Figure 2.

**BIG CLOUD FABRIC BENEFITS**

**Centralized Controller Reduces Management Consoles By Over 60:1**
With configuration, automation and most troubleshooting done via the BCF Controller, the number of management consoles involved in provisioning new physical capacity or new logical apps goes down dramatically. For example, in a 32 rack pod with dual leaf switches and four spine switches, a traditional box-by-box network design would have 68 switch management consoles. The Big Cloud Fabric design has only one—the controller console—that performs the same functions. The result is massive time savings, reduced error rates and simpler automation designs. As a powerful management tool, the controller console exposes a web-based GUI, a traditional networking-style CLI and REST APIs.

**Streamlined Configuration, Enabling Rapid Innovation**
In the BCF design, configuration in the CLI, GUI or REST API is based on the concept of logical tenants. Each tenant has administrative control over a logical L2/L3/policy design that connects the edge ports under the tenant’s control. The Big Cloud Fabric controller has the intelligence to translate the logical design into optimized entries in the forwarding tables of the spine, leaf and vleaf.

**Open Networking Switch Hardware Reduces CapEx Costs By Over 50%**
By adding up hardware, software, maintenance and optics/cables, a complete picture of the hard costs over three years shows that the savings are dramatic.

**Built-in Orchestration Support Streamlines DC Operations**
BCF Controller natively supports integration with various Cloud Management Platforms (CMPs)—VMware (vSphere, NSX Manager, vSAN, & VIO), and OpenStack, and Container orchestrators—through a single programmatic interface. This is tremendously simpler and scalable compared to box-by-box networking which demands an exponentially larger number of programmatic interactions with CMPs. Data center admins benefit from streamlined application deployment workflows, enhanced analytics and simplified troubleshooting across physical and virtual environments.

**SDN Fabric Enables Deep Visibility and Resilience for OpenStack Networks**
The BCF OpenStack Neutron plugin for L2/L3 networking provides resiliency necessary for production-grade OpenStack deployments—including support for distributed L3 routing and distributes NAT (Floating IP). The BCF Controller acts as the single pane for provisioning, troubleshooting, visibility and analytics of the entire physical and virtual network environment. This enables data center operators to deploy applications rapidly, simplifies operational workflows and provides immediate root-cause analysis when application performance issues arise.
Network / Security / Audit Workflow Integration

BCF Controller exposes a series of REST APIs used to integrate with application template and audit systems, starting with OpenStack. By integrating network L2 / L3 / policy provisioning with OpenStack HEAT templates in Horizon GUI, the time to deploy new applications is reduced dramatically as security reviews are done once (on a template) rather than many times (on every application). Connectivity, edit, and audit functions allow for self-service modifications and rapid audit-friendly reporting, ensuring efficient reviews for complex applications that go beyond the basic templates.

Scale-out (Elastic) Fabric

BCF’s flexible, scale-out design allows users to start at the size and scale that satisfies their immediate needs while future proofing their growth needs. By providing a choice of hardware and software solutions across the layers of the networking stack and pay-as-you-grow economics, starting small scale and growing the fabric gradually instead of locking into a fully integrated proprietary solution, provides a path to a modern data center network. Once new switches (physical or virtual) are added, the controller adds those switches to the fabric and extends the current configuration hence reducing any error that may happen otherwise. Customers take advantage of one-time configuration of the fabric.

DC-grade Resilience

BCF provides DC grade resiliency that allows the fabric to operate in the face of link or node failures as well as in the rare situation when the controller pair is unavailable (headless mode). Swapping a switch (in case of HW failure or switch repurpose) is similar to changing a line card in a modular chassis. After re-cabling and power up, the switch boots up by downloading the right image, configuration and forwarding tables. Additionally, the BCF Controller coordinates and orchestrates the entire fabric upgrade ensuring minimum fabric down time. These functionalities further enhance fabric resiliency and simplify operations.

USING BCF: A 3-TIER APPLICATION EXAMPLE

BCF supports a multi-tenant model, which is easily customizable for the specific requirements of different organizations and applications. This model increases the speed of application provisioning, simplifies configuration, and helps with analytics and troubleshooting. Some of the important terminology used to describe the functionality include:

- **Tenant** — A logical grouping of L2 and/or L3 networks and services.
- **Logical Segment** — An L2 network consisting of logical ports and end-points. This defines the default broadcast domain boundary.
- **Logical Router** — A tenant router providing routing and policy enforcement services for inter-segment, inter-tenant, and external networks.
- **External Core Router** — A physical router that provides connectivity between Pods within a data center and to the Internet.
- **Tenant Services** — Services available to tenants and deployed as dedicated or shared services (individually or as part of a service chain).
Tenant Workflow
In the most common scenario, end consumers or tenants of the data center infrastructure deal with a logical network topology that defines the connectivity and policy requirements of applications. As an illustrative example, the canonical 3-tier application in Figure 5 shows various workload nodes of a tenant named “BLUE.” Typically, a tenant provisions these workloads using orchestration software such as OpenStack, VMware vSphere, or BCF Controller GUI/CLI, directly. As part of that provisioning workflow, the BCF Controller seamlessly enables the logical topology onto the physical and virtual switches.

Mapping Logical to Physical
The BLUE Tenant has three logical network segments. Each of the three segments represents the broadcast domain for the 3-tiers—Web, App, and Database. Let’s say in this example, Web, App, and DB are virtualized workloads but DB is comprised of physical workloads. Following the rules defined by the data center administrator, the orchestration system provisions requested workloads across different physical nodes within the data center. As an example, the logical topology shown in Figure 5 could be mapped on the pod network. The BCF Controller handles the task of providing optimal connectivity, between these loads dispersed across the pod, while ensuring tenant separation and security.

In order to simplify the example, we only show racks that host virtualized and physical workloads in the figure above, but similar concepts apply for implementing tenant connectivity to external router and chaining shared services.
## Big Cloud Fabric Features

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description / Benefit</th>
</tr>
</thead>
</table>
| Zero Touch Fabric (ZTF) | ZTF enables complete control and management of physical switches within BCF without manually interacting with the switches. It tremendously simplifies day-to-day network operations:  
  • auto-configuration and auto-upgrade of Switch Light OS  
  • automatic topology updates and event notifications based on fabric link state changes  
  • auto-scaling of the fabric—adding or removing nodes and/or links within the fabric requires no additional configuration changes on the controller |
| Fabric Lag            | 1U Fabric LAG combines the underlying LAG functionality in switching ASICs with the centralized visibility of the SDN controller to create a highly resilient and efficiently balanced fabric. Compared to spanning tree protocols or even traditional MLAG/ECMP based approaches to multi-path fabric formation, Fabric LAG technology enables significantly reduced convergence time on topology changes and dramatically reduced configuration complexity. |
| Fabric Sync           | Fabric Sync intelligently synchronizes Controller Information Base (CIB) with fabric node’s Forwarding Information Base (FIB) using the OpenFlow protocol. During a topology change, only delta updates are synced across impacted switches. Fabric Sync ensures strong CIB-FIB consistency, as it is the single point of control for maintaining all forwarding and associated policy tables. |
| Resilient Headless Mode | In situations when both controllers are unreachable, fabric nodes are considered to be running in Headless mode. In this mode, all provisioned services continue to function as programmed prior to entering the Headless mode. Additionally, multiple levels of redundancy enable a highly resilient and self-healing fabric even during headless mode. |
| Centrally-managed Fabric (GUI, CLI & REST APIs) | Big Cloud Fabric Controller provides single pane of glass for entire fabric.  
  • Administrators can configure, manage, debug or troubleshoot, and upgrade the fabric nodes using CLI, GUI, or REST API.  
  • REST APIs, CLI and GUI have application and tenant awareness.  
  Single Pane of Glass fabric management enhances operational simplicity by providing a centralized dashboard for fabric management as well as quick and easy access to troubleshooting, analytics and telemetry information. Additionally, it provides simplified workflow for network operators and administrators. |
<p>| Fabric Analytics      | Fabric Analytics is the set of features that provides Advanced Multi-node Troubleshooting, Analytics &amp; Telemetry in the Big Cloud Fabric solution.                                                                                                                                     |
| API-first Fabric      | Big Cloud Fabric Controller is highly programmable due to its “API-first” design principle and can be implemented as a closed loop feedback system. For example, security applications can dynamically detect threats and program the BCF controller for mitigation. The BCF GUI and CLI utilize the underlying REST APIs—hence are by definition consistent and hardened. |
| Tenant-aware Fabric   | Big Cloud Fabric provides built-in multi-tenancy via tenant-aware configurations, tenant separation, and fine-grain inter-tenant access control. Configuration in the CLI, GUI or REST API is based on the concept of logical tenants. |
| Service-aware Fabric  | Big Cloud Fabric supports L3 virtual and physical service insertion and service chaining. Services can be shared across tenants or dedicated to a specific tenant.                                                                                                    |</p>
<table>
<thead>
<tr>
<th>FEATURE</th>
<th>DESCRIPTION / BENEFIT</th>
</tr>
</thead>
</table>
| **L2 Features** | • Layer 2 switch ports and VLAN trunks  
• IEEE 802.1Q VLAN encapsulation  
• Support for up to 4K VLANs (i.e. 4K Logical Segments)  
• MAC address based segmentation  
• BPDU Guard  
• Storm Control  
• MLAG (up to 16 ports per LAG)  
• 3,800 IGMP Groups  
• IGMP Snooping  
• Static Multicast Group  
• Link Layer Discovery Protocol (LLDP)  
• Link Aggregation Control Protocol (LACP): IEEE 802.1AX  
• LACP Fallback Mode (Dynamic membership management for server PXE booting)  
• Jumbo frames on all ports (up to 9216 bytes)  
• VLAN Translation  
• Primary / Backup Interface  
• VXLAN Support |
| **L3 Features** | • Layer 3 interfaces: Routed ports, Switch Virtual Interface (SVI), Distributed Gateway  
• Multiple IP-Subnet Support per Segment/SVI  
• Support for up to 46K IPv4 host prefix, 14K IPv6 host prefix (i.e. Endpoints)  
• Support for 1K Virtual Routing and Forwarding (VRF) entries (i.e. 1K Logical Routers)  
• 1K Tenants  
• Static Route, BGP (IPv4, IPv6), OSPF (IPv4)  
• 68K IPv4 routes, 8K IPv6 routes  
• Up to 16 ways Equal-Cost Multipathing (ECMP)  
• 1K Equal-Cost Multipathing (ECMP) groups  
• 3K flexible ACL entries  
• Policy-Based Routing  
• Multicast Routing  
• ACL: Routed ACL with Layer 3 and 4 options to match ingress ACL  
• Jumbo frame support (up to 9216 bytes)  
• DHCP relay  
• NAT/PAT support |
| **QoS**       | • Layer 2 IEEE 802.1p (class of service [CoS])  
• Source segment or IP DSCP based Classification  
• Tenant/Segment based classification  
• DWRR based egress queuing  
• CoS based marking  
• PFC and DCBX  
• IP address/subnet based QoS classification |
<table>
<thead>
<tr>
<th>FEATURE</th>
<th>DESCRIPTION / BENEFIT</th>
</tr>
</thead>
</table>
| High Availability     | • Controller HA  
• Headless mode (fabric forwards traffic in absence of Controller)  
• Redundant Spine  
• Redundant Leaf  
• Redundant Links  
• Controller cluster with single Virtual IP  
• Support for redundant out-of-band management switch |
| Security              | • Ingress ACLs  
• Layer 3 and 4 ACLs: IPv4, Internet Control Message Protocol (ICMP), Transmission Control Protocol (TCP), User Datagram Protocol (UDP), etc.  
• ACLs on controller management interface  
• ACL logging (IPv4 only)  
• Control Plane Policing (CoPP) or Rate Limiting  
• Custom TLS keys and Certs support for GUI  
• EAP-TTLS Support for Radius Authentication  
• Two Factor Authentication  
• Restrict Cipher Suites  
• FIPS 140-2 compatibility |
| OpenStack Integration | • Nova-network & Neutron ML2 Driver mechanism support  
• Neutron L3 (IPv4/IPv6) Plugin support (distributed routing, Floating IP, PAT and Security Group visibility)  
• Auto Host Detection & LAG Formation  
• OpenStack Horizon Enhancements (Heat Templates, Fabric Reachability Test)  
• Dynamic Provisioning of the BCF Fabric  
• Distributed Routing and NAT  
• Tenant driven OpenStack Router policy configuration grid  
• Multiple Logical Router support within an OpenStack project  
• LBaaS support (network automation driven through OpenStack)  
• SR-IOV Integration |
| VMware vSphere Integration | Provides Fabric Automation and Visibility including:  
• Auto Host Detection & LAG Formation  
• Auto L2 Network Creation & VM Learning  
• Network Policy Migration for vMotion/DRS  
• VM-level Visibility (VMname, vMotion)  
• VM-to-VM Troubleshooting (Logical & Physical)  
• L3 configuration via vSphere web-client plugin  
• Test Path visibility through vCenter  
• Multiple tenants per vCenter |
| VMware NSX-v Support  | Close the overlay/underlay gap for visibility and troubleshooting. Features include:  
• Auto Host Detection & LAG Formation  
• Auto Network Creation for VTEP Port-Group & VTEP Discovery  
• Underlay Troubleshooting - VTEP-to-VTEP connectivity  
• Underlay Visibility through Fabric Analytics (VM-name, VXLAN ID, Logical Switch)  
• NSX Hardware VTEP support |
<table>
<thead>
<tr>
<th>FEATURE</th>
<th>DESCRIPTION / BENEFIT</th>
</tr>
</thead>
</table>
| VMware vSAN Support | • Auto-detection and LAG formation for vSAN node  
                             • Auto-creation of vSAN transport network  
                             • vSAN cluster communication troubleshooting for unicast and multicast  
                             • Simplified Layer 2 / Layer 3 multicast deployment for vSAN transport  
                             • vSAN Analytics                                                                                                                                  |
| Nutanix Integration | • Automatic Host Bootstrapping  
                             • Auto Host Detection & LAG Formation (support all AHV load-balancing modes)  
                             • Auto L2 Network Creation & VM Learning  
                             • Auto L3 Network Creation & Distributed Logical Routing  
                             • Network Policy Migration for VM Migrations  
                             • AHV Networking and VM-level Visibility (VMname, Host Information, Physical Connectivity)  
                             • VM-to-VM Troubleshooting with Test-Path (Logical & Physical)  
                             • Multiple tenants per AHV Cluster                                                                                                                 |
| Container Integration | • Container Network Interface (CNI) plugin support  
                             • Auto Host Detection & LAG Formation  
                             • Dynamic Provisioning of the BCF Fabric  
                             • Network isolation for container virtual cluster / namespace  
                             • Container-level visibility (Container name, virtual switch, IP/MAC, vNIC)  
                             • Container-to-container troubleshooting  
                             • K8s/OpenShift service visibility on BCF Fabric Analytics                                                                                         |
| Multi-Orchestration Support | Support Multiple Virtual PODs (vPODs) on single BCF Fabric                                                                                                                                                       |
| Inter-Pod Connectivity | • L3 — Using Static Route, OSPF & BGP  
                             • L2 — Dark Fiber  
                             • L2 — VXLAN  
                             • Hub and Spoke topology (scale)                                                                                                                     |
| MIBs                | Documented in a separate MIBs document                                                                                                                                                                          |
| Industry Standards  | • IEEE 802.1p: CoS prioritization  
                             • IEEE 802.1Q: VLAN tagging  
                             • IEEE 802.3: Ethernet  
                             • IEEE 802.3ae: 10 Gigabit Ethernet  
                             • IEEE 802.3ba: 40 Gigabit Ethernet                                                                                                                  |
### Support for Open Networking Ethernet Switching

Support Broadcom Trident-II+, Tomahawk & Maverick ASICs for 10G, 25G, 40G and 100G switches from Dell, HPE and Accton / EdgeCore. The common supported switch configurations are:

- 48x10G + 6x40G
- 48x10GT + 6x40G
- 48x10G + 6x40G/4x100G
- 48x10GT + 6x40G/4x100G
- 32x40G
- 48x25G + 6x100G
- 64x40G
- 32x100G

For the complete list of supported switch vendors/configurations as well as optics/cables, included in the Big Cloud Fabric Hardware Compatibility List (HCL), please contact the Big Switch Sales Team at sales@bigswitch.com.

### Fabric Management

- **GUI (IPv4 / IPv6)**
- **CLI (IPv4 / IPv6)** — based console to provide detailed out-of-band management
- **REST API (IPv4 / IPv6)**
- Switch management using 10/100/1000-Mbps management through controller
- **Beacon LED** (based on underlying switch)
- **Configuration synchronization**
- **Configuration save and restore**
- **Secure Shell Version 2 (SSHv2)** — IPv4 / IPv6
- **Username and passwords authentication**
- **TACACS+ / RADIUS** — IPv4 / IPv6
- **Control Plane Security (CPSec)** — Encrypted communication between Controllers and Physical / Virtual Switches
- **Syslog** (4 servers) — IPv4 / IPv6
- **SNMP v1, v2c and v3** — IPv4 / IPv6
- **sFlow support**
- **SPAN with Policy/ACL**
- **Fabric SPAN with Policy/ACL**
- **Connected device visibility**
- **Ingress and egress packet counters per interface, per segment, and per tenant**
- **Network Time Protocol (NTP)** — IPv4 / IPv6
- **Test Path** — Enhanced Troubleshooting & Visibility with logical and physical fabric views (VM <-> vLeaf <-> Leaf <-> Spine <-> Leaf <-> vLeaf <-> VM)
- **Fabric Analytics** including telemetry and enhanced analysis
BCF CONTROLLER APPLIANCE SPECIFICATION

The BCF Controller can be deployed either as a physical appliance (production or lab deployment) or as a virtual machine appliance (for limited scale production or lab deployment). Physical appliance is also available in NEBS form factor.

BCF Controller — Physical Appliance Specification: The BCF controller is available as enterprise-class, 2-sockets, 1U rack-mount physical appliance designed to deliver the right combination of performance, redundancy and value in a dense chassis. It comes in two versions – Standard and Large.

<table>
<thead>
<tr>
<th>FEATURE</th>
<th>TECHNICAL SPECIFICATION*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Controller</td>
<td>HWB (Standard)</td>
</tr>
<tr>
<td>Recommended for</td>
<td>BCF P-fabric</td>
</tr>
<tr>
<td></td>
<td>(up to 16 racks)</td>
</tr>
<tr>
<td></td>
<td>BCF P+V fabric</td>
</tr>
<tr>
<td></td>
<td>BCF P-fabric</td>
</tr>
<tr>
<td></td>
<td>(more than 16 racks)</td>
</tr>
<tr>
<td></td>
<td>BCF P+V fabric</td>
</tr>
<tr>
<td></td>
<td>BCF P-fabric</td>
</tr>
<tr>
<td></td>
<td>(more than 16 racks)</td>
</tr>
<tr>
<td></td>
<td>BCF P or</td>
</tr>
<tr>
<td></td>
<td>P+V Fabric</td>
</tr>
<tr>
<td>Processor</td>
<td>Intel Xeon 2 sockets</td>
</tr>
<tr>
<td></td>
<td>(6 / 8 cores)</td>
</tr>
<tr>
<td></td>
<td>Intel Xeon 2 sockets</td>
</tr>
<tr>
<td></td>
<td>(12 cores)</td>
</tr>
<tr>
<td></td>
<td>Intel Xeon 2 sockets</td>
</tr>
<tr>
<td></td>
<td>(12 cores)</td>
</tr>
<tr>
<td></td>
<td>Intel Xeon 2 sockets</td>
</tr>
<tr>
<td></td>
<td>(10 cores)</td>
</tr>
<tr>
<td>Form Factor</td>
<td>1U Rack Server</td>
</tr>
<tr>
<td>Memory</td>
<td>4 x 16GB</td>
</tr>
<tr>
<td>Hard Drive</td>
<td>2 x 1TB SATA</td>
</tr>
<tr>
<td></td>
<td>(w/RAID support)</td>
</tr>
<tr>
<td>Networking</td>
<td>4 x 1Gb; 2 x 10Gb</td>
</tr>
<tr>
<td>Power</td>
<td>Dual Hot-plug power</td>
</tr>
<tr>
<td></td>
<td>supply 500W/550 W</td>
</tr>
<tr>
<td></td>
<td>Dual Hot-plug power</td>
</tr>
<tr>
<td></td>
<td>supply 500W/550 W</td>
</tr>
<tr>
<td></td>
<td>Dual Hot-plug power</td>
</tr>
<tr>
<td></td>
<td>supply DC 1100 W</td>
</tr>
<tr>
<td></td>
<td>Dual Hot-plug power</td>
</tr>
<tr>
<td></td>
<td>supply 550W</td>
</tr>
</tbody>
</table>

* Detailed environment information provided in BCF Hardware Guide.

VM APPLIANCE SPECIFICATION

The Big Cloud Fabric Controller is available as a Virtual Machine appliance for P or P+V fabric (for limited scale production or lab deployment).

<table>
<thead>
<tr>
<th>ENVIRONMENT</th>
<th>LAB ONLY</th>
<th>PRODUCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux KVM</td>
<td>Ubuntu 14.04</td>
<td></td>
</tr>
<tr>
<td>VMware ESXi</td>
<td>Version 6.0, 6.5</td>
<td>Version 6.5</td>
</tr>
<tr>
<td>Red Hat RHEL</td>
<td>RHEL 7.2, 7.4</td>
<td>RHEL 7.2, 7.4</td>
</tr>
<tr>
<td>vCPU</td>
<td>10 vCPU</td>
<td>12 vCPU</td>
</tr>
<tr>
<td>vMemory</td>
<td>56 GB of Virtual Memory</td>
<td>56 GB of Virtual Memory</td>
</tr>
<tr>
<td>HDD</td>
<td>300GB HDD</td>
<td>300GB HDD</td>
</tr>
<tr>
<td>vNIC</td>
<td>4 vNICs</td>
<td>4 vNICs</td>
</tr>
</tbody>
</table>

Note: A VM’s performance depends on many other factors in the hypervisor setup, and as such, we recommend using a hardware appliance for production deployments greater than three racks.
### SUPPORTED WORKLOADS & ORCHESTRATION SYSTEMS

<table>
<thead>
<tr>
<th>FEATURE</th>
<th>TECHNICAL SPECIFICATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Physical Workloads</td>
<td>Bare-metal server workloads</td>
</tr>
<tr>
<td>Virtual Workloads</td>
<td>VMware Integration with vSphere 6.0, 6.5.</td>
</tr>
<tr>
<td></td>
<td>For OpenStack see table below.</td>
</tr>
<tr>
<td></td>
<td>VMware Horizon View VDI</td>
</tr>
<tr>
<td></td>
<td>For container integration see table below.</td>
</tr>
<tr>
<td></td>
<td>Support any VM workload on BCF P Fabric even without orchestration integration (e.g. Xen, Hadoop).</td>
</tr>
<tr>
<td>Cloud Orchestration</td>
<td>OpenStack (Neutron ML2 driver, Neutron L3 Plugin)</td>
</tr>
<tr>
<td></td>
<td>VMware VIO</td>
</tr>
<tr>
<td></td>
<td>Container Integration—Container Network Interface (CNI) plugin support</td>
</tr>
</tbody>
</table>

### OPENSTACK INTEGRATION

<table>
<thead>
<tr>
<th>HYPervisor</th>
<th>OPENSTACK – PIKE</th>
<th>OPENSTACK – OCATA</th>
<th>OPENSTACK – NEWTON</th>
</tr>
</thead>
<tbody>
<tr>
<td>KVM</td>
<td>CentOS 7.4 (Packstack)</td>
<td>CentOS 7.4 (Packstack)</td>
<td>CentOS 7.3, 7.4 (Packstack)</td>
</tr>
<tr>
<td>RHEL 7.4 (RHOSP 12)</td>
<td>RHEL 7.4 (RHOSP 11)</td>
<td>RHEL 7.3, 7.4 (RHOSP 10)</td>
<td></td>
</tr>
</tbody>
</table>

### CONTAINER INTEGRATION

<table>
<thead>
<tr>
<th>DISTRO</th>
<th>CONTAINER ORCHESTRATION</th>
<th>CONTAINER RUNTIME VERSION</th>
<th>BARE METAL OS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kubernetes Community</td>
<td>Kubernetes 1.9 - 1.10</td>
<td>Docker 1.13</td>
<td>CentOS 7.4 / Ubuntu 16.04</td>
</tr>
<tr>
<td>Red Hat OpenShift 3.5</td>
<td>Kubernetes 1.5</td>
<td>Docker 1.12</td>
<td>RHEL 7.4</td>
</tr>
<tr>
<td>Red Hat OpenShift 3.9</td>
<td>Kubernetes 1.9</td>
<td>Docker 1.13</td>
<td>RHEL 7.4</td>
</tr>
</tbody>
</table>