#Pivotal Customer[0] *PCF on vSphere* Reference Architecture

Goal: Customer[0] Reference Architectures are utilized by Pivotal's Customer[0] group to simulate a base deployment of our products that is common to as many customer use cases as possible. These architectures are then automated via concourse pipelines and validated thru various Customer[0] validation scenarios to simulate typical customer use cases.

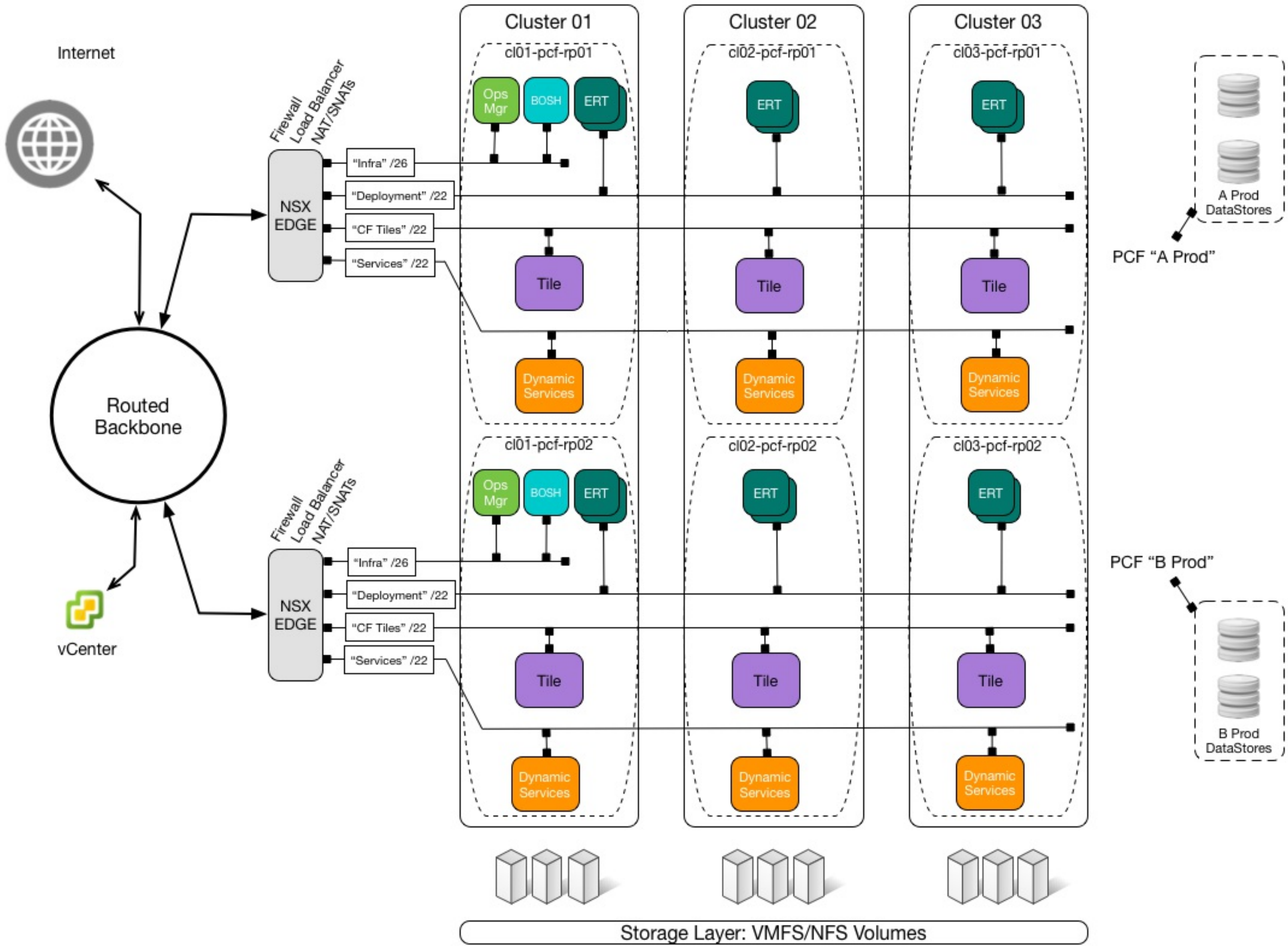
Customer[0] Typical Customer = *A secured and internally hosted PCF Foundation, capable of hosting ~1500 Application Instances (AIs) with PCF managed Services: "MySQL, RabbitMQ, Pivotal Spring Cloud Services"*

Non-Goals:

- This PCF on vSphere reference architecture is published as is with no warranty or support expressed or implied.
- This document is not intended to replace the basic installation documentation located @ docs.pivotal.io, but rather to demonstrate how those instructions should be related to a recommended Pivotal Cloud Foundry installation.

PCF Products Validated	Version
PCF Ops Manager	1.9.latest
Elastic Runtime	1.9.latest

Pivotal Customer[0] Reference Architecture Overview



- Pipeline Repo Link : Customer[0] Concourse Pipelines
- Pipeline ERT Repo Link : Customer[0] Concourse Pipelines
- Running Pipeline Link : See the Running Customer[0] Concourse Pipelines

The reference approach is to create three Clusters, populate them with the Resource Pools and then deploy PCF with Pivotal Ops Manager into those pools, one pool per cluster. Every AZ in PCF maps to a resource pool in a cluster, never the cluster itself, to provide resource separation and a organizational model aligned to the installation.

Core networking is created via an NSX Edge with the following subnets:

- Infrastructure
- ERT (*Elastic Runtime*)

- Service tiles (one or more)
- Dynamic service tiles (a network managed entirely by BOSH Director)

This model is the gold standard for deploying one or more PCF installations for long term use and growth, allowing for capacity growth at the vSphere level, app capacity growth in PCF and also maximum installation security.

This diagram shows two PCF installations sharing the same vSphere capacity, yet segmented from each other with Resource Pools (the dotted line rectangles). This approach can easily scale to many PCF installations on the same capacity with the assurance that each is resource protected and separate from each other. Priority can be given to one or another installation if desired thru the use of "shares" applied at the pool level ("High Shares" for the important installation, "Low Shares" for the sacrificial one(s)).

Compute:

Each Cluster is populated by a minimum of three ESXi hosts, making nine hosts for each PCF installation in a stripped manner. All installations tap the capacity of the same nine hosts in an aggregated fashion. Vertical growth is accomplished thru adding more pools and PCF installations, horizontal growth is via adding more hosts to the existing clusters (in threes, one host per Cluster), from which all the installations can gain access to the added capacity.

It is a VMware best practice to deploy hosts in Clusters of no less that three for vSphere HA use. vSphere DRS is a required function to enable Resource Pools and allow for automated vMotion.

Storage:

Storage is granted to the hosts in one of two common approaches:

1. Datastores are granted to all hosts and a subset are offered to one installation at a time.
2. Datastores are granted to a host cluster uniquely and each installation uses multiple datastores to store VMs per cluster.

Example (1): There are 6 datastores, "ds01" thru "ds06". All nine hosts are granted access to all six datastores. PCF installation #1 is provisioned to use "ds01", "ds02", and "ds03" and VMs land in all the pools starting in "ds01" until it's full, then "ds02" is used and so on.

Example (2): There are 6 datastores, "ds01" thru "ds06". Cluster 1 hosts are granted "ds01" and "ds02", Cluster 2 hosts are granted "ds03" and "ds04", and so on. PCF installation #1 is provisioned to use "ds01", "ds03" and "ds05" and all VMs land on the datastore correct for the cluster they are provisioned to. This is how vSphere VSAN works.

Datastore sizing is recommended at 8 TB per installation, or smaller volumes that aggregate up to this quantity. Small installations that won't have many tiles added can use less, 4 TB per PCF is reasonable. The primary consumer of storage is the NFS/WebDav blobstore.

As of this publication (reference PCF versions above), PCF does not support the use of vSphere Storage Clusters. Datastores should be listed individually in the vSphere tile.

If a vSphere datastore is part of a vSphere Storage Cluster using sDRS (storage DRS), the sDRS feature must be disabled on the datastores used by PCF as s-vMotion activity will cause BOSH to malfunction as a result of renaming managed independent disks.

Recommended types of storage are block-based (fiber channel or iSCSI) and file-based (NFS) over high speed carriers such as 8Gb FC or 10GigE. Redundant storage is highly recommended for the "persistent" storage type used by PCF. DASD or JBOD can be used for the "ephemeral" storage type.

Networking

The above model employs VMware NSX SDN (software-defined networking) to provide unique benefits to the PCF installation on vSphere. Refer to subsequent chapters in this document for treatments of this approach where NSX is not used.

The use of NSX is an optional, but highly recommended, addition to the installation approach, as it adds many powerful elements:

1. Distributed firewall capability per-installation thru the built-in Edge Firewall
2. High capacity, resilient, distributed load balancing per-installation thru the NSX Load Balancer
3. Installation obfuscation thru the use of non-routed RFC-1918 networks behind the NSX Edge and the use of SNAT/DNAT connections to expose only the endpoints of Cloud Foundry that need exposure.
4. High repeatability of installations thru the repeat use of all network and addressing conventions on the right hand side of the diagram (the Tenant Side)
5. Automatic rule and ACL sharing via NSX Manager Global Ruleset
6. Automatic HA pairing of NSX Edges, managed by NSX Manager
7. Support for PCF Go Router IP membership in the NSX Edge virtual load balancer pool by the BOSH CPI (not an Ops Manager feature)

NSX DLR (Distributed Logical Router) is not used in this approach as it provides only routing services, not load balancing and firewalling.

NSX DLF (Distributed Logical Firewall) isn't used as we gain that capability right where it's needed most, in front of the PCF installation, not on the network(s) the installation uses. This also applies to "micro-segmentation", as there's no need to place firewall rules horizontally on a network used by PCF when the above model is deployed with the NSX Edge.

NSX DLB (Distributed Load Balancing) isn't used as it's not considered production quality and is not intended for use with L7 balancing (which is PCF's primary need) or for North/South flows.

Networking Design

Each PCF installation consumes four (or more) networks with the NSX Edge, aligned to specific jobs:

- "Infrastructure": A network with a small CIDR range for use with those resources focused on interacting with the IaaS layer and back-office systems. This is an "inward-facing" network, where Ops Manager, BOSH ad other utility VMs such as jump box VM would connect.
- "Deployment": A network with a large CIDR range exclusively used by the ERT tile to deploy app containers and related support components. Also known as "the apps wire".
- "CF Tiles": At least one, if not more, with a large CIDR range for use with other installations hosted and managed by BOSH via Ops Manager. A simple approach is to use this network for all PCF tiles except ERT. A more involved approach would be to deploy multiple "Services-#" networks, one for each tile or one for each type of tile, say databases vs message busses and so on.
- "Dynamic Services": A single network granted to BOSH Director for use with service tiles that require dynamic address space to deploy

onto. This is the only network that will be marked as "Services" with a check box in the vSphere tile.

All of these networks are considered "inside" or "tenant-side" networks, and use non-routable RFC-1918 network space to make provisioning repeatable. The NSX Edge translates between the tenant and service provider side networks using SNAT and DNAT.

Each NSX Edge should be provisioned with at least four service provider side (routable) IPs:

1. A static IP by which NSX Manager will manage the NSX Edge
2. A static IP for use as egress SNAT (traffic from tenant side will exit the Edge on this IP)
3. A static IP for DNATs to Ops Manager

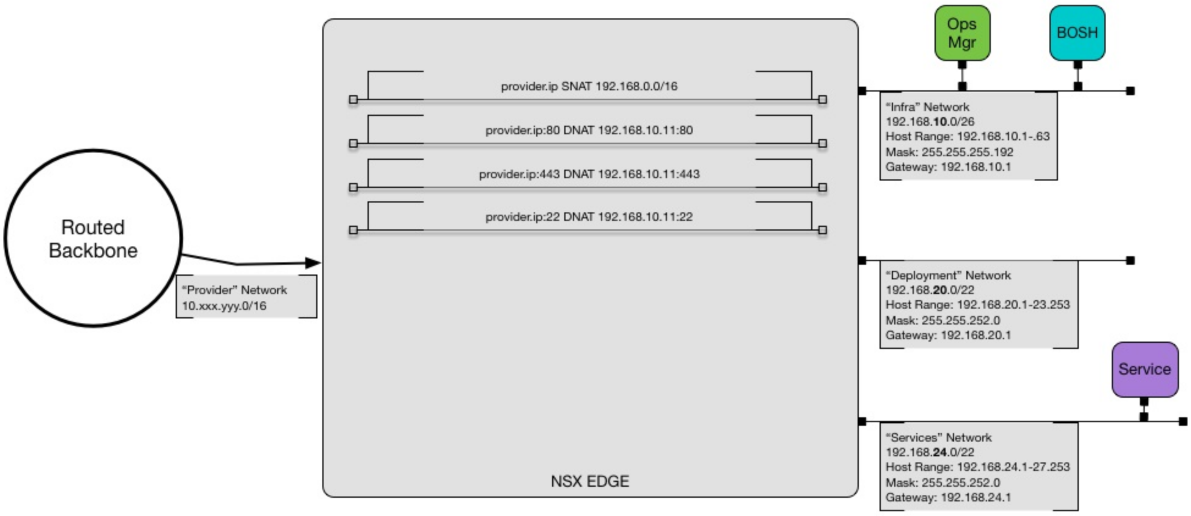
4. A static IP for the load balancer VIP that will balance to a pool of PCF GoRouters (HTTP/HTTPS)

   There are many more uses for IPs on the routed side of the Edge. Ten reserved, contiguous static IPs are recommended per NSX Edge for flexibility and future needs.

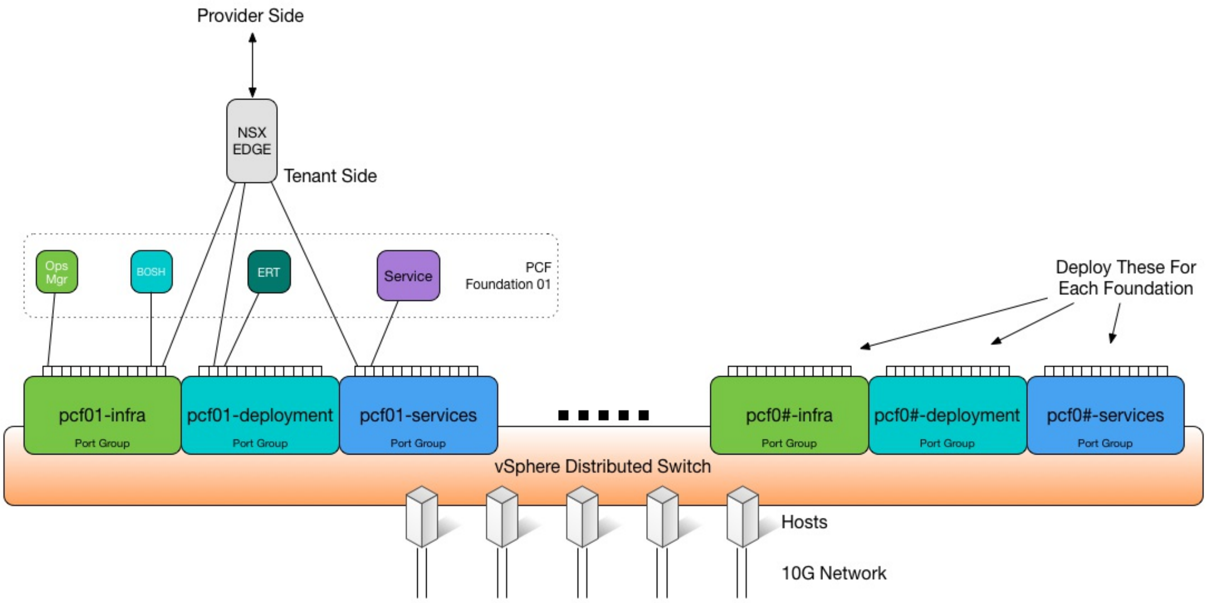
On the tenant side, each interface on the Edge that is defines will act as the IP gateway for the network used on that port group. The following are recommend for use on these networks:

- "Infra" network: 192.168.10.0/26, Gateway at .1
- "Deployment" network: 192.168.20.0/22, Gateway at .1
- "CF Tiles" network: 192.168.24.0/22, Gateway at .1
- "Dynamic Services" network: 192.168.28.0/22, Gateway at .1

- *Future Use: "Services-B" network: 192.168.32.0/22, and so on...*



vSphere DVS (Distributed Virtual Switching) is recommended for all Clusters used by PCF. NSX will create a DPG (distributed port group) for each interface provisioned on the NSX Edge. Alternatively, NSX Logical Switches can be used on the Tenant Side of this design, which leverages vWires, reducing the dependency on VLAN address space.



Reference Approach Without VMware NSX

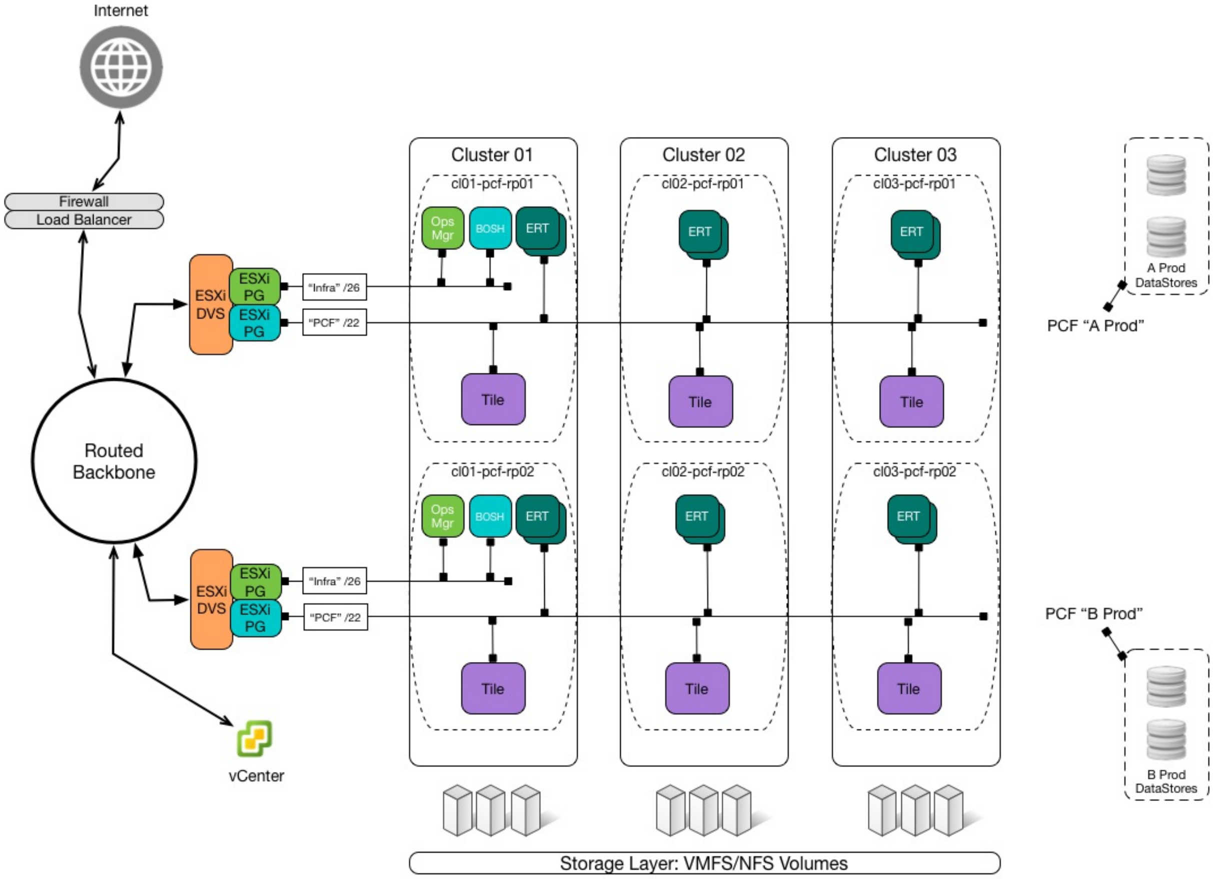
In the absence of VMware NSX SDN technology, the PCF installation on vSphere follows the standard approach discussed in the documentation. For the purposes of this reference architecture, it would be easiest to explore what changes and/or is lost in this approach.

Networking Features

- Load balancing would have to be hosted by some external service, such as a hardware appliance or VM from a 3rd party. This also applies to SSL termination.
- Per-installation firewalling would be lost, as the traditional approach to firewalling inside systems is per zone or per network, not per virtual appliance installation that spans multiple networks.
- The need to SNAT/DNAT non-routable RFC-1918 networks used with PCF would go away as it's unlikely they would be used at all without the NSX Edge there to provide the boundary. In it's place a single, or possible multiple VLANs from the routable network space already deployed in the datacenter would be used.

Networking Design

The more traditional approach without SDN would be to deploy a single VLAN for use with all of PCF, or possibly a pair of VLANs (one for infrastructure and one for PCF). As VLAN capacity is frequently limited and scarce, this design seeks to limit the need for VLANs to a functional minimum.



In this example, the functions of firewall and load balancer have been moved outside the of vSphere space to generic devices assumed to be available in the datacenter. The PCF installation is now bound to two port groups provided by a DVS on ESXi, each one aligned to a key use case:

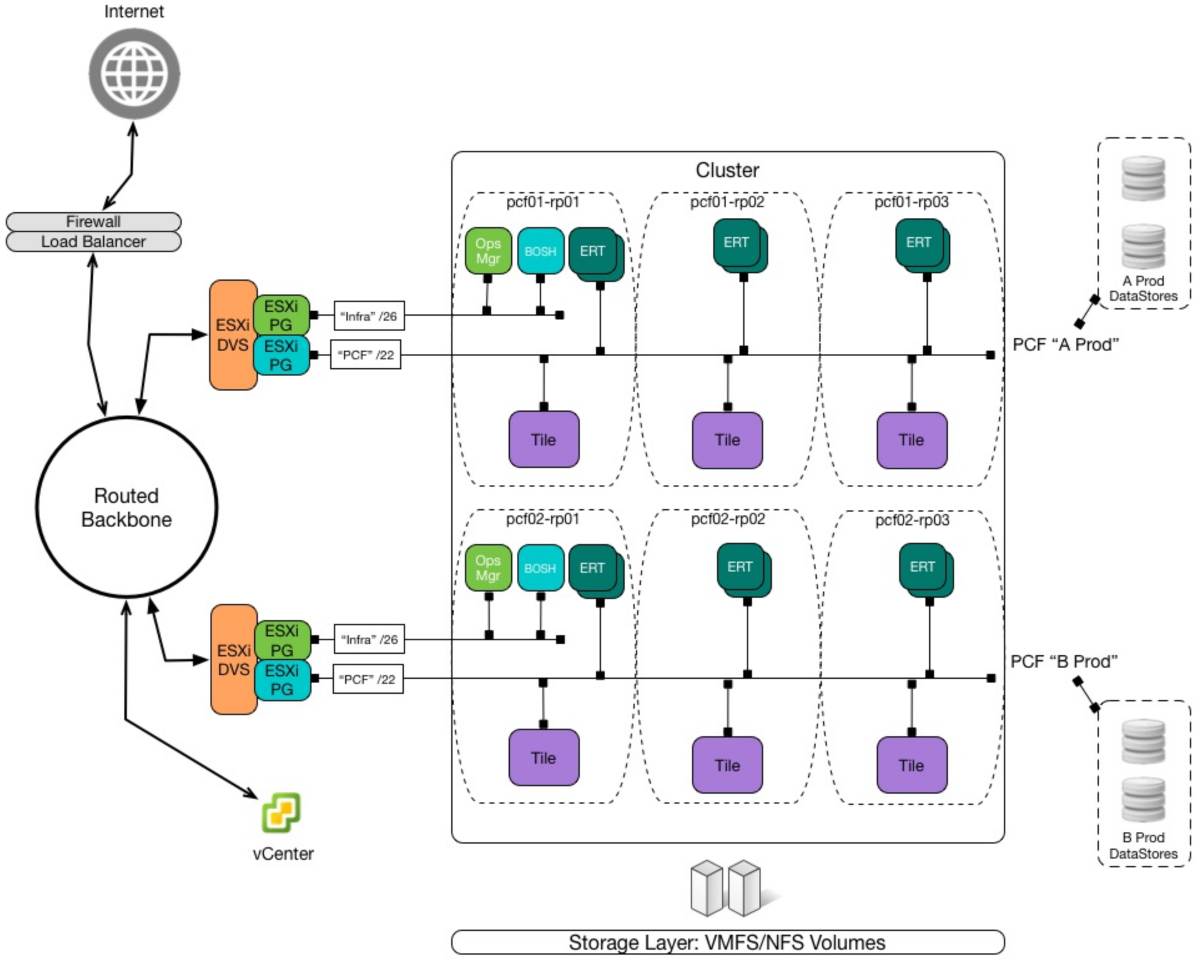
1. "Infra": PCF VMs used to communicate w/the IaaS layer
2. "PCF": the primary deployment network for all tiles including ERT

Each one of these port groups typically is assigned a VLAN out of the datacenter's pool and a routable IP address segment. Routing functions are handled by switching layers outside of vSphere, such as TOR or EOR switch/router.

It's still valid to deploy all the networks shown in the original design, so if the resources are readily available, feel free. The main thing to keep in mind is that this is a requirement per PCF installation, so keep a count of how many of those overall you will require.

Reference Approach Without Three Clusters

Some desire to start with PCF aligned to fewer resources than the standard (above) calls for, so the starting point for that is a single Cluster. If you are working with at least three ESXi hosts, the recommended guidance is still to setup in three Clusters, even with one host in each (such that the HA comes from the PasS, not the IaaS), but for less than that, place all available hosts into a single Cluster with DRS and HA enabled.



A two Cluster configuration has little value compared to a single or triple cluster configuration. While a pair of Clusters has symmetry in vSphere, PCF always seeks to deploy resources in odd numbers, so a two Cluster configuration forces the operator into a two AZ alignment for odd (three) elements, which is far from ideal.

Network Design

It is recommended to use the networking approach detailed in either the with-NSX or without-NSX sections for this design, as the compute arrangement has little impact on how PCF is networked for production use.

Storage

It is recommended that all datastores to be used by PCF be mapped to all the hosts in the single cluster. Otherwise, follow the guidance from above.

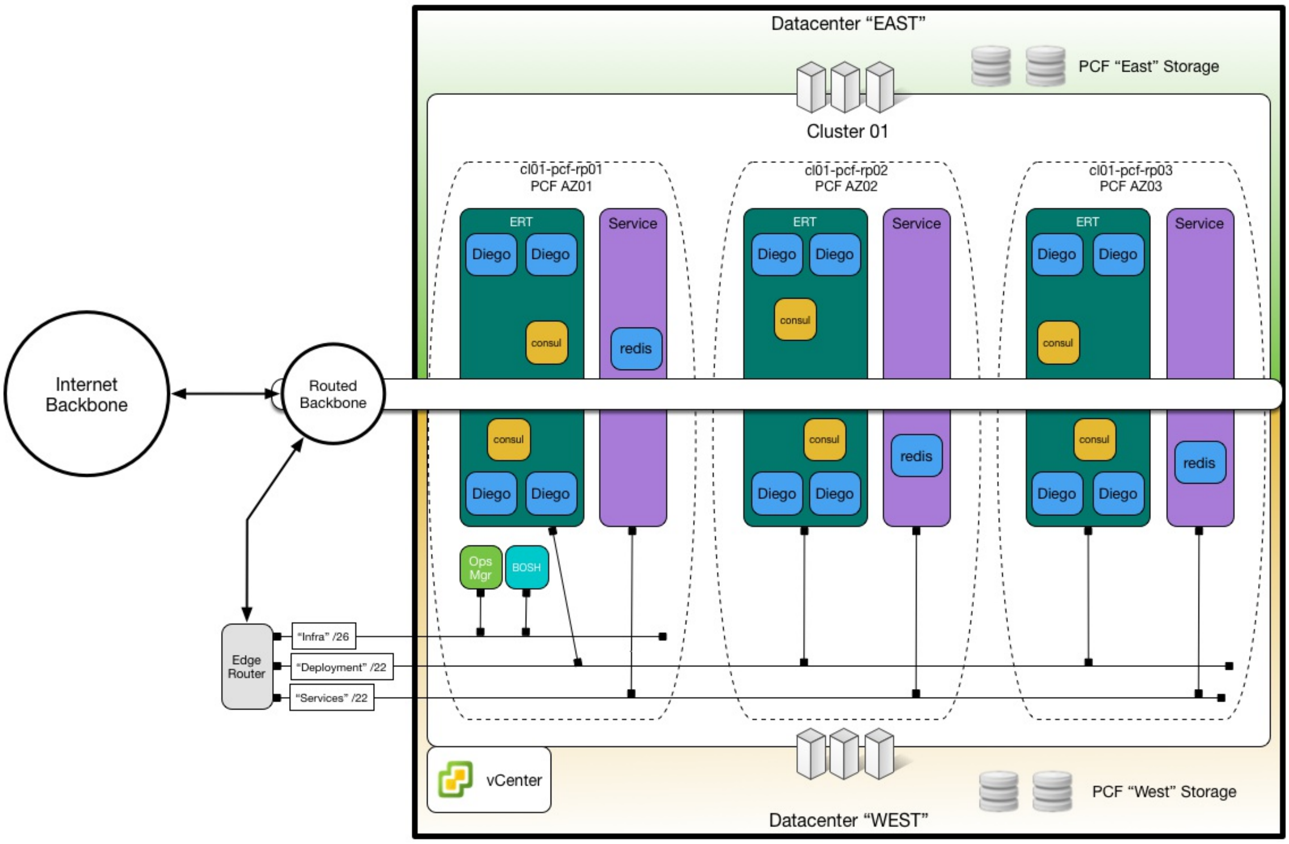
Reference Approach Utilizing Multi-Datacenter

For some scenarios, deploying PCF across combined resources located in more than one site is desirable to avoid total loss of site. There are a number of approaches architects may take to solve this problem, each with it's own caveats.

TL;DR PCF Multi-Datacenter is a plausible approach that's flawed in one way or another depending on the architecture.

Multi-Datacenter vSphere With Stretched Clusters

In this approach, the architect is treating two sites as the same logical capacity and is building Clusters from components from both sites at the same time. Given four hosts, two might come from "East" and two might come from "West". In vSphere, these appear to form a four host Cluster. Networking is applied such that all hosts see the same networks thru stretched layer 2 application or perhaps a SDN solution such as NSX is being used to tunnel L2 over L3.



In terms of PCF, the Cluster is an AZ and BOSH has no sense of some of that capacity coming from different places. Thus, these hosts must be able to operate such that there's no practical difference between the networks and storage they attach to, in terms of latency and connectivity.

To honor the (above) gold standard, the approach should be three Clusters in use, but drawing from two sites, yielding a 4x3x3 type of deployment.

- Four hosts per cluster (two from each site)
- Three clusters for PCF as AZs
- Three AZs mapped to Clusters in PCF

Also, the single cluster model (above) can be used. This may be the more practical approach, since so many resources from both sites are already being applied to achieve HA.

Replicated storage between sites is assumed. Datastores must be common to all hosts in a cluster for seamless operation, or else VMs will become trapped on the hosts mapped to specific datastores and won't vMotion away for maintenance or move for DRS.

An interesting strategy for this model to ensure high availability for PCF is to keep a record of how many hosts are in a cluster and deploy enough copies of a PCF job in that AZ to ensure survivability in a site loss. This means placing large, odd numbers of jobs (such as consul) in the cluster so that at least two are left on either site in the event of a loss of site. In a four host cluster, this would call for five consul job VMs, so each site has at least two if not the third. DRS anti-affinity rules can be used here (set at the IaaS level) to force like VMs apart for best effect.

Also, lots of smaller Diego Cells are recommended over a few, very large Diego Cells.

Network traffic is a challenge in this scenario, as app traffic may enter at any point in either site's connection points, but can only leave at a designated gateway. Thus, it's possible to have apps servicing traffic coming from either East or West but only appearing to respond via West (as in the diagram) causing a "trombone effect" of traffic doubling across datacenter links. The architect should consider the impact of hosting apps that may land in East only to have the traffic flow out of West.

Multi-Datacenter vSphere With Combined East/West Clusters

In this approach, the architect is drawing capacity from the two sites independently and offering clusters of this capacity to PCF in distinct sets. This could yield vSphere Clusters always in pairs (East & West), so honoring PCF's need to deploy in odd numbers can become problematic.

One strategy here would be to effectively double the standard approach at the beginning of this material, yielding six total clusters, three from each side. While this seems like a whole lot of gear to apply to PCF, you could argue that in a BC/DR type of scenario, doubling everything is exactly the point.

Another strategy is to use the Single Cluster approach from above, where you have three resource pools in one cluster per site, yielding six AZs to PCF but only using one actual cluster of capacity from each site. This approach won't scale as readily but does have the benefit of drawing capacity from only one cluster, which is east to provision with only a few hosts.

Storage replication in this case is less critical as the assumption is there are enough AZs from either side to survive a failure and vSphere HA isn't needed to recover the installation.