Emerging IT technologies have brought about a shift from IT as a cost center to IT as a business driver. Business relevance has regained its proper center stage spotlight. The majority of this change has been brought about by cloud operating models. These models have brought with them the concept of software-defined networking (SDN) and related technologies, which in turn moved the network into focus as the final major component for transition. That transition is to put the application as the primary focus.

Applications

Applications influence business. They include web portals that generate revenue, human resource (HR) systems that help to on-board new employees, imaging systems that support patient care, etc. While the group of applications used differs between industry and between businesses within an industry, one thing remains constant: an application is not an isolated process running on a virtual machine (VM). Applications are ecosystems of interconnected components - some physical, some virtual, some legacy, some new.

These application ecosystems are complex interconnections of components and tiers, the end result of which promotes business value. For example, a doctor may access an MRI on a tablet while consulting with a patient. The process, from entering the patient information to receiving the image, may be complete in seconds. On the backend, a web-based portal provides the search form, passing the doctor’s credentials and the patient’s identification to an authorization system. Once the doctor is authorized as a valid user the information is passed to a mid-range or UNIX system for a patient records search. Next, this information is passed to an indexing system of image storage, which then retrieves the MRI image from a network file system (NFS) storage device. Finally, this information is formatted and compiled through a presentation layer and presented back to the doctor in the patient room.
This interaction to provide a single image to a doctor interacting with a patient requires several tiers and components spread across both front-end and data center networks. These application components may exist in virtual or physical layers and the transactions are most likely subject to various network services, including security and compliance checks. Figure 1 shows an example.

**Figure 1. Example Application View**

As shown in Figure 1, applications are not as simple as software running on a VM. The intricacies within application connectivity go well beyond the scope of Figure 1 when TCP/UDP ports, service chaining, and more are added. This complete ecosystem is what provides the desired end result to the user or system using the application.

### Current Network Definition of Applications

While real-world applications look like the ones described above, today's networks do not treat them that way. Today's networks group applications by virtual LAN (VLAN) and subnet and apply connectivity and policy based on those constructs. This leads to restrictions on how applications can be grouped and how policy can be applied to those applications.

One or more applications are grouped into VLANs and then IP subnets are mapped to those VLANs. From there connectivity through routing is configured and network services are applied to the subnet addressing. Figure 2 illustrates this relationship.
The coupling shown in Figure 2 is not conducive to mapping applications onto the network or to applying policies to those applications once there. The current model lends itself to misconfiguration, and policy configuration which is looser than desired. Manual configurations, process, and coupled constructs lead to slower deployment, higher configuration error rates, and reduced auditability. This creates significant business impact with the current methodology.

ACI Endpoint Groups

ACI Endpoint Groups (EPGs) provide a new model for mapping applications to the network. Rather than using forwarding constructs such as addressing or VLANs to apply connectivity and policy, EPGs use a grouping of application endpoints. EPGs act as a container for collections of applications, or application components and tiers that can be used to apply forwarding and policy logic. They allow the separation of network policy, security, and forwarding from addressing and instead apply it to logical application boundaries. Figure 3 provides an example EPG.
Figure 3. End-Point Groups (EPG.)

While simplistic, Figure 3 shows a grouping of HTTP and HTTPS services as a single group of endpoints known as an EPG. This grouping is independent of addressing, VLAN, and other network constructs as opposed to traditional network environments that must rely on these for groupings.

Within an EPG separate endpoints can exist in one or more subnets, and subnets could be applied to one or more EPGs based on several other design considerations. Layer 2 forwarding behavior can then be applied independently of the Layer 3 addressing. Figure 4 shows the relationship between EPGs and subnets.

Figure 4. Relationship Between EPGs and Subnets

Figure 4 shows two subnets being applied to two different services within an EPG. HTTPS endpoints reside in 10.10.11.x while HTTP endpoints reside in 10.10.10.x. Regardless of the separate subnets, policy is applied to both HTTPS and HTTP services within this EPG in this example. This illustrates the complete decoupling of addressing from policy enforcement. Within the ACI fabric, subnets are utilized for forwarding only and policy can be enforced granularly within a subnet, or consistently across subnets.

Endpoint groups not only allow for better mapping of applications to the network itself, but also for better mapping of the network to application owners and developers. Rather than application owners and developers being required to maintain mappings to IP addressing and subnets they can group applications or application components to logical EPGs.
EPGs are designed as flexible containers for endpoints that require common policy. Several methods exist for defining endpoints and placing them in EPGs. Once grouped, policy is applied based on the logical grouping rather than addressing and forwarding constructs. The use of EPGs can and will differ across customer environments and even across a single fabric deployment.

**ACI Application Network Profiles**

ACI Application Network Profiles are the grouping of one or more EPGs and the policies that define how they communicate. These are used to define the connectivity of application tiers such as web-app-database, compute, network-storage. They are sometimes thought of as connectivity graphs. Application Network Profiles are the instantiation of a complete application on the network. These profiles are how network engineers, application owners, and developers map an application onto the network hardware. Figure 5 illustrates this relationship.

**Figure 5. Application Network Profiles**

In Figure 5 an Application Network Profile is shown as a group of three EPGs and the policies that define how each communicates. This is a simplistic example of how a typical application would appear on the network. EPGs group common components of a complete application. These EPGs are grouped with the communication and policies between them to define the entire application as an Application Network Profile.

The combination of EPGs and Application Network Profiles allows for truly stateless network policy definition and enforcement; this enforcement is completely free from dependencies on locality and forwarding. Application policy is defined in the user space and automatically mapped onto the network hardware when applied based on workload location.

While it is beyond the scope of this document to discuss Application Network Profiles in great detail, it is important to understand the basics. For more information refer to the [APIC Policy Model White Paper](#).

**EPG Usage**

EPGs are designed to abstract the instantiation of network policy and forwarding from basic network constructs (VLANs and subnets.) This allows applications to be deployed on the network in a model consistent with their development and intent. Endpoints assigned to an EPG can be defined in several ways. Endpoints can be defined by virtual port, physical port, IP address, DNS name, and in the future through identification methods such as IP address plus Layer 4 port and others.
There is no dedicated manner in which EPGs should be deployed and utilized; instead the remainder of this document will cover some typical uses for EPGs. Examples include:

- Mapping traditional network constructs to the ACI fabric
  - EPG as VLAN
  - EPG as a subnet (model classic networking using EPGs)
  - EPG as virtual extensible LAN (VXLAN)/Network Virtualization using Generic Routing Encapsulation (NVGRE) virtual network identifier (VNID)
  - EPG as a VMware port group
- Utilizing the ACI fabric for stateless network abstraction
  - EPG as an application component group (web, app, database, etc.)
  - EPG as a development phase (development, test, production)
  - EPG as a zone (internal, DMZ, shared services, etc.)

### Mapping Traditional Network Constructs to the ACI Fabric

The following four sections provide examples for mapping existing applications onto the ACI fabric using traditionally defined network constructs. Topics covered include EPG as a VLAN; EPG as a subnet (model classic networking using EPGs); EPG as VxLAN/NvGRE VNID; and EPG as a VMware port group.

#### EPG as a VLAN

The EPG as a VLAN method is useful for both an initial mapping of existing applications onto the ACI fabric and for incorporating existing systems in mixed environments. In this model the VLAN and all of the devices within that VLAN are mapped into an EPG.

This method can be utilized both logically for migrating applications and physically when connecting to existing network infrastructure. In the logical usage the devices attached to an existing VLAN are defined as members of a single EPG within the ACI fabric. Once the logical configuration is in place, the actual devices, virtual or physical, can be migrated onto the ACI fabric. Figure 6 depicts using EPG as a VLAN as a logical migration tool.

**Figure 6.** Using EPG as a VLAN to Map Applications onto the ACI Fabric
In Figure 6 applications are migrated from existing networks by assigning endpoints from existing VLANs into EPGs. After the logical configuration is complete the endpoints can be migrated onto the ACI fabric. This is one possible migration method for existing applications.

This use of EPG as a VLAN is primarily for integrating existing network infrastructure with the ACI fabric. This could be existing switches, including blade switches, which are already in use within the data center. Using EPG as a VLAN for integration is shown in Figure 7.

**Figure 7.** Using EPG as a VLAN for Integration

![Figure 7](image.png)

Figure 7 shows existing network equipment attached to the ACI fabric through leaf switches. In this scenario existing VLANs will be mapped into EPGs upon fabric ingress. Once endpoints are assigned to the EPG, policy is applied based on the EPG assignment rather than the VLAN. This method can be used to attach existing blade chassis or other switches or networks to the fabric. This method can be used as both a migration path and a permanent or semi-permanent integration solution.

**EPG as a Subnet**

EPG as a subnet is another method for mapping applications onto the ACI fabric in a method mirroring traditional networking. Rather than redesigning the application layout for a given application, existing subnets can be configured as EPGs. All devices in the assigned IP address range will become members of the EPG and receive consistent policy.

This model will fall in line with many current service appliance (firewall, application delivery controller [ADC], etc.) deployment models which utilize the IP subnet to identify and apply policy to traffic. Policy will be applied based on the EPG which is equal to the original subnet. Additionally, this model allows for a quick and straightforward migration to the ACI fabric. Figure 8 depicts the mapping of subnets to ACI fabric EPGs.
This EPG usage is very similar to the VLAN as an EPG method described above. The key difference is that within the ACI fabric Layer 2 semantics such as flooding are only enabled if required. This difference reduces unnecessary traffic for devices that only require Layer 3 communication.

EPG as a subnet mirrors application deployment on a classic network. This model is most useful for migrating existing applications onto the ACI fabric in the same manner in which they are currently deployed. This method is also useful for connecting to outside networks such as the WAN. It is important to note that EPG usage models are not mutually exclusive. This means that an EPG as a subnet model can be used for existing applications while new applications can be deployed, gaining more benefit from other EPG usage models.

**EPG as a VXLAN VNID/NVGRE Video Source ID (VSID)**

Many virtualized environments have moved toward overlay models which utilize VXLAN or NVGRE for tunneling. This tunneling allows virtual machine connectivity independent of the underlying network. In these environments one or more virtual networks are built using the chosen overlay technology and traffic is encapsulated as it traverses the physical network.

The ACI fabric is designed to provide overlay independence and can bridge frames to and from VXLAN, NVGRE, VLAN, and 802.1x encapsulation. This provides flexibility for heterogeneous environments which may have services residing on disparate overlays.

The virtual networks or VNIDs created by these overlays can be translated directly into EPGs within the ACI fabric. This provides a method for quickly migrating onto the ACI fabric as well as providing policy instantiation for these networks in production use. Figure 9 shows the use of VXLAN and NVGRE tunnels as an EPG.
Figure 9. VXLAN and NVGRE Tunnels as EPGs

Figure 9 shows the mapping of both a VXLAN and NVGRE virtual network to EPGs within the ACI fabric. This can be configured to happen automatically at the ingress leaf, translating all traffic from a given VNID into an EPG at ingress. What is not shown in this example is the ability to route traffic between these disparate overlays within the fabric, if desired.

EPG as a VXLAN VNID or NVGRE VSID is a powerful tool for both migration and virtual connectivity in environments requiring greater scale than that provided by VLANs. VLANs are limited at a theoretical 4096 application tenants while both NVGRE and VXLAN provide for more than 16 million separate virtual networks. Additionally, overlapping IP subnets can be used for these separate virtual networks if they are configured as separate fabric tenants or separate, private Layer 3 domains within a tenant.

EPG as a VMware Port Group

This model of EPG use is very similar to the EPG as a VLAN method as the connectivity from the ACI leaf perspective is still VLAN-based. The difference is that the configuration and integration of the VLAN constructs on the VMware Distributed Virtual Switch (DVS) within the VMware environment are automated by the Cisco® Application Policy Infrastructure Controller (APIC). Both Microsoft HyperV and Linux-based hypervisors rely on standard VLAN configuration for switching, and would typically be configured using EPG as a VLAN for existing workloads being moved to the ACI fabric.

A DVS mapped to the ACI environment is created, uplinks (physical network interface cards, known as pNICs) are added to the construct, and port groups are created. Each port group receives a user-friendly name and either a VLAN, a trunk (multiple VLANs), or is left untagged. VMs are provided connectivity by assigning their virtual NICs (vNIC) to a specific port group. This is outlined in Figure 11.
Figure 10. VMware vSwitch and DVS Configuration

Figure 10 depicts VMs connected to a DVS using port groups.

In the model, EPG as a VMware port group, the ACI fabric will be configured to translate VMware port group assignment into EPG assignment. This will be done by applying VLAN tags at the port-group level, which will be interpreted by the leaf and translated into EPGs in much the same way as EPG as a VLAN above.

This mode can be configured manually or performed through integration between the fabric and VMware. VMware passes VM traffic from the hypervisor with VLAN tags assigned per port group. These tags are then translated into EPGs at the ACI leaf switch and policy is applied. Figure 11 shows EPG as a VMware port group.

Figure 11. EPG as a VMware Port Group

Figure 11 shows the mapping of port groups to ACI EPGs through the leaf. The translation would occur based on the VLAN assigned to the VMware port group, which would be carried in the 802.1q header.
Utilizing the ACI Fabric for Policy Instantiation and Service Insertion

The following three sections provide examples of the way in which EPGs can be used to provide policy instantiation and service insertion by removing ties to traditional constructs such as VLAN and subnet. These examples include EPG as an application component group (web, application, database, etc.); EPG as a development phase (development, test, production); and EPG as a zone (internal, DMZ, shared services, etc.). It is important to note that EPG usage methods are not mutually exclusive and can be utilized on a per-application or EPG basis.

EPG as an Application Component Group

EPG as an application component group is the primary basis of EPG design. In this method EPGs are designated as logical groups of endpoints that represent a specific component or tier of an application. For example, an EPG may represent the endpoints serving as the web portal of a multi-tier application.

This model will typically align most closely to the design driven by application architects. Additionally, this model frees application architects from needing knowledge of underlying constructs such as VLANs and subnets. Using EPGs to designate application component groups allows for clear policy application between tiers in a fashion organic to the way in which they are designed.

In this model EPGs can be thought of as having a provider/consumer relationship with one another where the communication between them is defined by policy contracts. This allows for both simple and complex applications to be designed logically with actual network instantiation handled by the fabric automatically. Figure 12 illustrates this model.

Figure 12. EPG as an Application Component Group
EPG as a Development Phase

EPG as a development phase is another advanced model enabled by the ACI fabric. This method is designed to help expedite the development, test, and release of new software. The ACI fabric can help to expedite this process by allowing the various stages to coexist, securely isolated on the same fabric. Furthermore, the environments can coexist with or without overlapping IP space. These development tiers can then be seamlessly promoted from one stage to another, or rolled back if necessary.

Utilizing EPG as a development phase, testing and deployment tiers can be assigned to individual EPGs. Policy is then applied uniquely to each tier as required. A unique advantage here is that while separate tiers may use separate resources and network services, the policy definition itself can be configured once and applied to each tier in an identical fashion. This helps to prevent mistakes based on test and production environments with different security and service configurations than the development environment. Figure 13 highlights the use of EPG as a development phase.

**Figure 13.** EPG as a Development Phase

Figure 13 shows a three-phase development process deployed as three EPGs. The figure also shows the ability to push the endpoints from one phase to another by either moving the endpoints or changing the policy contracts. Using this method consistent development and test cycles can be performed without the need for network migration or policy change.

Because the definition of these development phases is not based on constructs such as VLAN and subnet, no addressing changes are required as these EPGs are pushed through the various development tiers. These typical network constructs can remain static and unchanged while the policy applied changes. This is a unique advantage of the ACI fabric.
EPG as a Zone

Utilizing EPG as a zone allows resource grouping based on security or compliance rules without the need to segregate these resources by VLAN or subnet. Examples of zones include DMZ, internal, shared services, PCI, HIPAA, and others. This allows for a logical segregation of devices that require specific security and compliance policies to be applied.

EPG as a zone alleviates the traditional network reliance on addressing and VLAN for segregation of resources and application of policy. Using this method allows developers and network teams to more closely coordinate policy and definition without the need for translation between common terminologies. It also decreases the complexity of the design and application of those defined policies.

Configuring EPG as a zone is done by grouping endpoints that require specific policy, such as DMZ and internal, into separate EPGs. Communication can then be created between these zones and other EPGs through defined policy contracts. Figure 14 illustrates the use of EPG as a zone.

Figure 14. EPG as a Zone

Figure 14 depicts the use of EPGs for zone segregation. In this example, separate EPGs are used for WAN connectivity (known as outside), DMZ resources, and internal resources. Specific contracts are defined to apply policy to the communication between these EPGs. It is important to note that within the ACI fabric communication is disallowed by default. This means that if no communication contract is applied, communication between EPGs will not occur. This simplifies configuration, reduces required rules, and increases the overall security of the fabric.
Summary

The ACI fabric is designed to abstract the complexities of network constructs and provide functionality in a format consumable by both network engineers and application owners. EPGs are the heart of this abstraction, allowing policy definition freed from addressing, routing, and VLAN constructs.

EPGs are designed to be used as a flexible container for grouping the components of an application logically. There are no restrictions on the number of methods used within a fabric, or the combination of EPG grouping usage within an application network profile. Figure 15 shows the use of multiple grouping methods within an application network profile.

Figure 15. Combining EPG Usages within an Application Network Profile

The use of EPGs is intended to be extremely flexible in order to allow the right fit for the right task, rather than a one-size-fits-all methodology. The examples described in this document are a subset of the way in which EPGs can be utilized. None of these methods is mutually exclusive, therefore different approaches can be taken for different applications or environments.

Additional References

For more information, read the Cisco ACI Policy Model (WP).