Cisco VMS Service Instantiation

The key differentiator of Cisco VMS is the orchestration and management of services using a deterministic and repeatable method, resulting in the consistent instantiation of a service. VMS, through the use of the service packs creates a consistent and well-formed service request, is able to instantiate a service based a well-defined service model and associated execution code. Each instantiated service will share common feature configuration and service topology.

This chapter contains the following sections:

Service Blueprints

The service pack services are made available at the Service Interface as a 'blueprint' or service definition model of an end-to-end service. For example, the service interface portal may offer a blueprint for a Virtual Router service connected to a CPE device. A service definition model 'blueprint' is essentially a set of intellectual property developed to render a customized service that is intended to operate over physical and virtual infrastructure. That intellectual property is referred to as a software service pack.

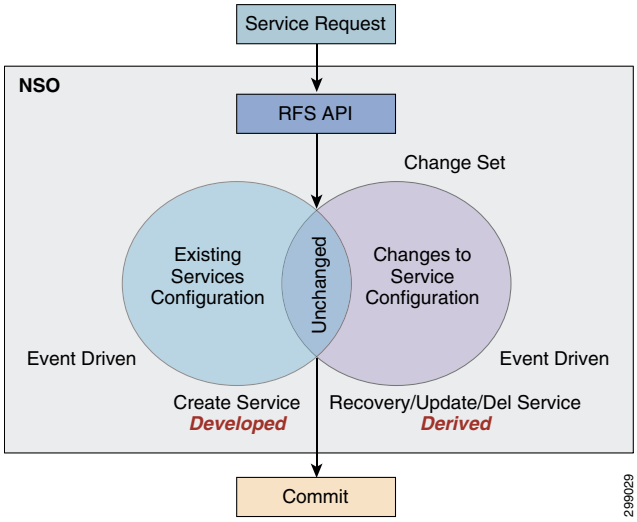
A software service pack lives in the Cisco NSO software modules and has several subcomponents. These subcomponents are the Service Model, Mapping Code, Device Model, and Network Element Drivers (NEDs). All of these components are required to instantiate service intention. When the elements of a service pack are compiled by the service developer, Cisco VMS solution automatically creates the APIs necessary for a northbound application (such as a portal) to request a given service definition model 'blueprint'.

Create, Read, Update, and Delete Configuration Optimizations

Create, Read, Update, Delete (CRUD) operations are at the heart of any services orchestration system. Typically these are achieved through somewhat complex workflow design. The Cisco NSO orchestrator software models provide a unique approach to solving this complexity issue. Hardened software processes enabled by Tail-F, service developers are only required to write the service create functions. Cisco NSO software automatically calculates all functions necessary to carry out the Read, Update, and Delete functions.

In Figure 13, a Venn diagram shows this operations concept. When a service request is made, Cisco NSO software will examine the requested service against any existing service currently deployed. A change set is then determined that represents the delta between the two service model definitions in the transaction database (CDB). Cisco NSO is capable of deriving all the actions necessary to move that new change of the service or device model set into operation. The transactional database (CDB) software allows the change sets to be unrolled either automatically if a service fails, or through operator request back to a previous stable state.

Figure 13 Cisco NSO Service Request Functionality



Service Device Mapping

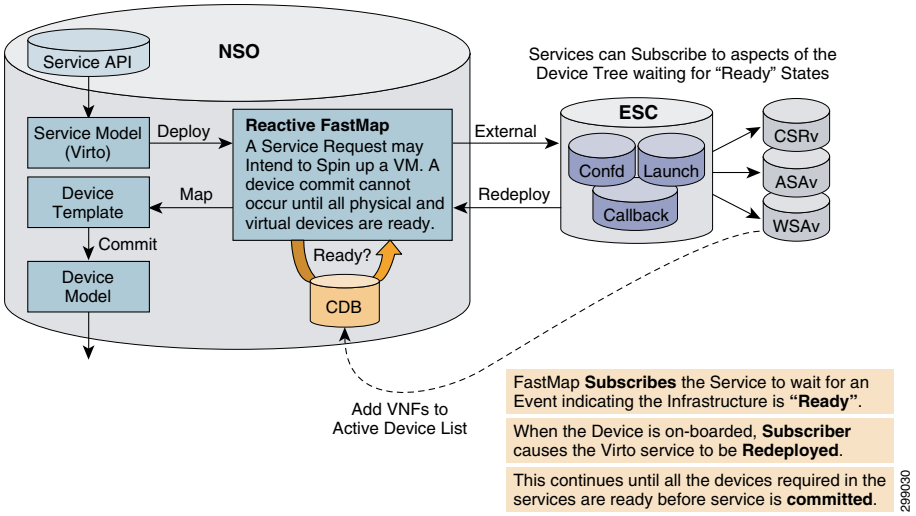
The purpose of the Fastmap software in the service pack of a service blueprint is to map service intent to the device infrastructure. The Fastmap software process uses Java logic resulting in the creation of a service template that is mapped to a device model. This process works very well for physical infrastructure. But what if the infrastructure does not yet exist, as is the case with virtual machines (VMs) running virtual network functions (VNFs). In these conditions, Cisco NSO can make use of the Reactive Fastmap software process.

The Reactive Fastmap software is capable of detecting when a service model requires a virtual network function (VNF) in the requested service model. Cisco NSO cannot complete service model mapping until all devices, both physical and virtual, are active. However, the Transactional Database (CDB) software cannot remain locked while the virtual devices are started.

In the case of the required VNF, NSO will call the Elastic Service Controller (ESC) software modules to handle the VNF life-cycle management. The Fastmap software process subscribes the service to an event in the transaction database indicating a VNF has been started and brought under management. When this occurs, Cisco NSO software will attempt to redeploy the service model requests. Here the entire service request process begins again with Cisco NSO software checking that all devices are in a ready state. If all physical and virtual devices are not ready, the Cisco NSO software modules will defer the service request again.

Eventually either the service deployment will fail because all devices cannot be brought into a ready state or all required components become fully available. In the case where the devices do not all appear in the ready state, the service request fails and all potential configurations are rolled back. When Cisco NSO detects that all devices are in a ready state, the service request process will proceed to map the service model to the appropriate device model. This entire process is shown in Figure 14.

Figure 14 Mapping Services to Non-Existing Infrastructure, Innovative Fastmap Functionality (Cloud VPN)



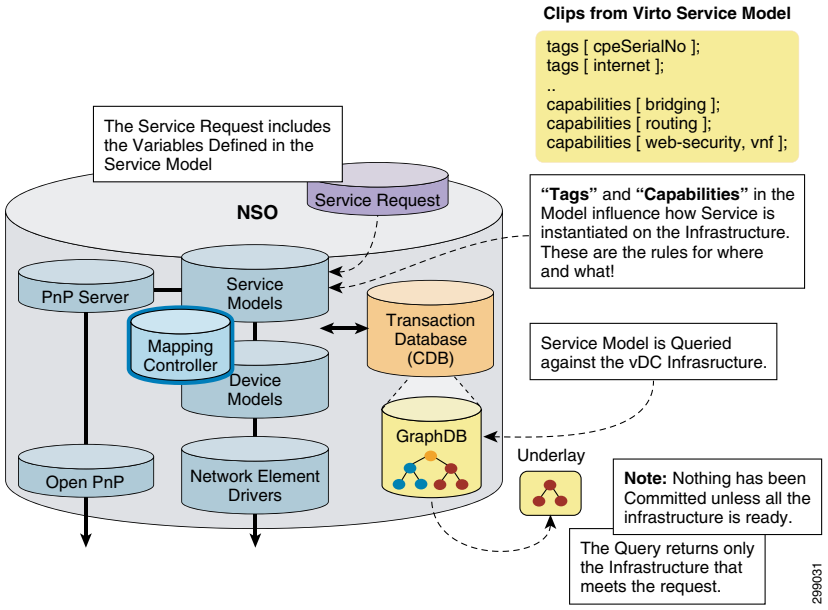
Configuring Infrastructure Elements

An underlay represents the physical and virtual infrastructure to which Cloud VPN services will be orchestrated. An underlay consists of physical and virtual devices, links, network bridges, and resource pools. Cisco NSO software is capable of supporting the loading of multiple infrastructure topologies into the transactional database (CDB).

The entire infrastructures is typically not necessary to instantiate every service. A service typically will require a subset of a given infrastructure. Cisco NSO software implements a mechanism known as the GraphDB, which is a tree representation of all the elements in the infrastructure. GraphDB software in Cisco NSO allows the Fastmap software processes to use queries based on the service model to find the applicable infrastructure required for the service.

To influence the result of the GraphDB query, Tags and Capabilities service requirements are programmed into the service request. Tags provide a description in the service model to which the Fastmap can parse that provides data to the mapping logic. Similarly, capabilities in the service definition model describe what is required of the infrastructure components for that particular service. The process of querying the GraphDB software modules is illustrated in Figure 15.

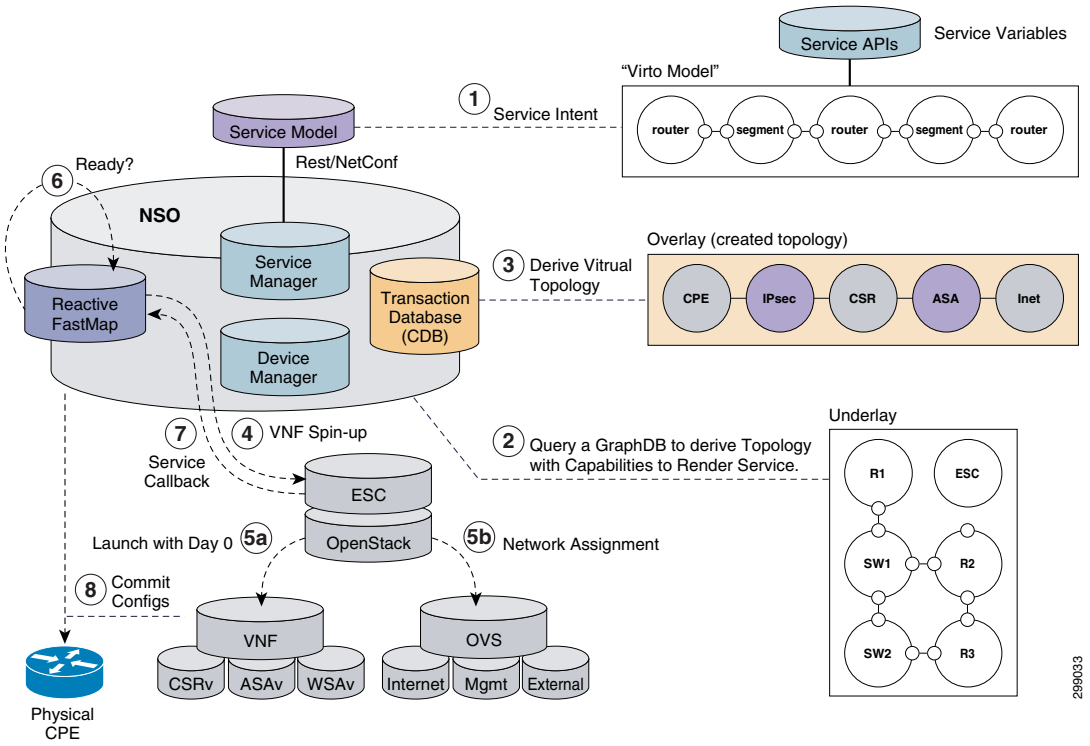
Figure 15 Mapping Service Intent to Specific Topology, Querying the Underlay



Putting the Pieces Together

Figure 16 is a complete representation of the advanced orchestration processes required to on-board a service. The first step is to ensure the Service Models required have been loaded into the Cisco NSO transaction database (CDB) The service model configurations that make up the service pack are written specifically to offer the required service. A service request received by Cisco NSO software will result in configuration checks against the service model which is handed off to the Fastmap process in Step 2. At this point GraphDB software is queried to retrieve only the infrastructure necessary to realize the specific service. Tags and Capabilities parameters in the service request will influence the service query. The output of the query request is the Service Overlay illustrated in Step 3. Based on the completion of these steps, Cisco NSO software will attempt to instantiate and create a Cloud VPN service comprised of both physical and virtual components.

Figure 16 Service Model Consumption



The reactive Fastmap software in Cisco NSO makes an external call to the Elastic Service Controller (ESC) software to spin up the required VMs in Step 4. ESC understands the affinity rules for launching the service, working with OpenStack VIM software (Nova, Neutron) to launch the necessary compute resources and bring up networking interfaces that enable the VMs to be manageable (Step 5). Cisco NSO software enables the initial minimal configurations of the newly launched VMs that are required to connect the VNFs to the management channel of Cisco NSO software.

During the process of launching the VMs, Cisco NSO software will release control of the database for other functions and wait (Step 6) until the VMs appear to be in the ready state. As with a CPE, a VM is online and ready once it completes a online is service call back process (Step 7) to put the newly started device into the Cisco NSO transactional database. Cisco NSO software will attempt to redeploy the service each time a service callback is made. However, no changes will be committed until all devices in the model register as ready in the transactional database (CDB).

Cisco NSO attempts to reach the physical and virtual components in the service overlay via SSH connectivity over the management network interface (Step 8). The service model configuration applied by Cisco NSO is referred to as the Day 1 configuration. When all configurations to all physical and virtual devices are complete then the specific service is considered deployed. If any of the configurations fail, then the service and device model changes applied in the Fastmap or Reactive Map processes are rolled back to the last known working configurations in the transaction database (CDB).