Pipeline
Technology for Service Providers.
www.pipelinepub.com Volume 14, Issue 2

# A CTO's Thoughts on OSS Modernization

By: David Deng

All communications service providers (CSP) have accumulated operational support systems (OSS) as they have built out their networks and services over their years of operation. A CSP’s OSS ecosystem may look like an ever-growing and tangling spider web as depicted in Figure 1, where many siloed systems are woven together across OSS, network management system (NMS), and element management system (EMS) layers.

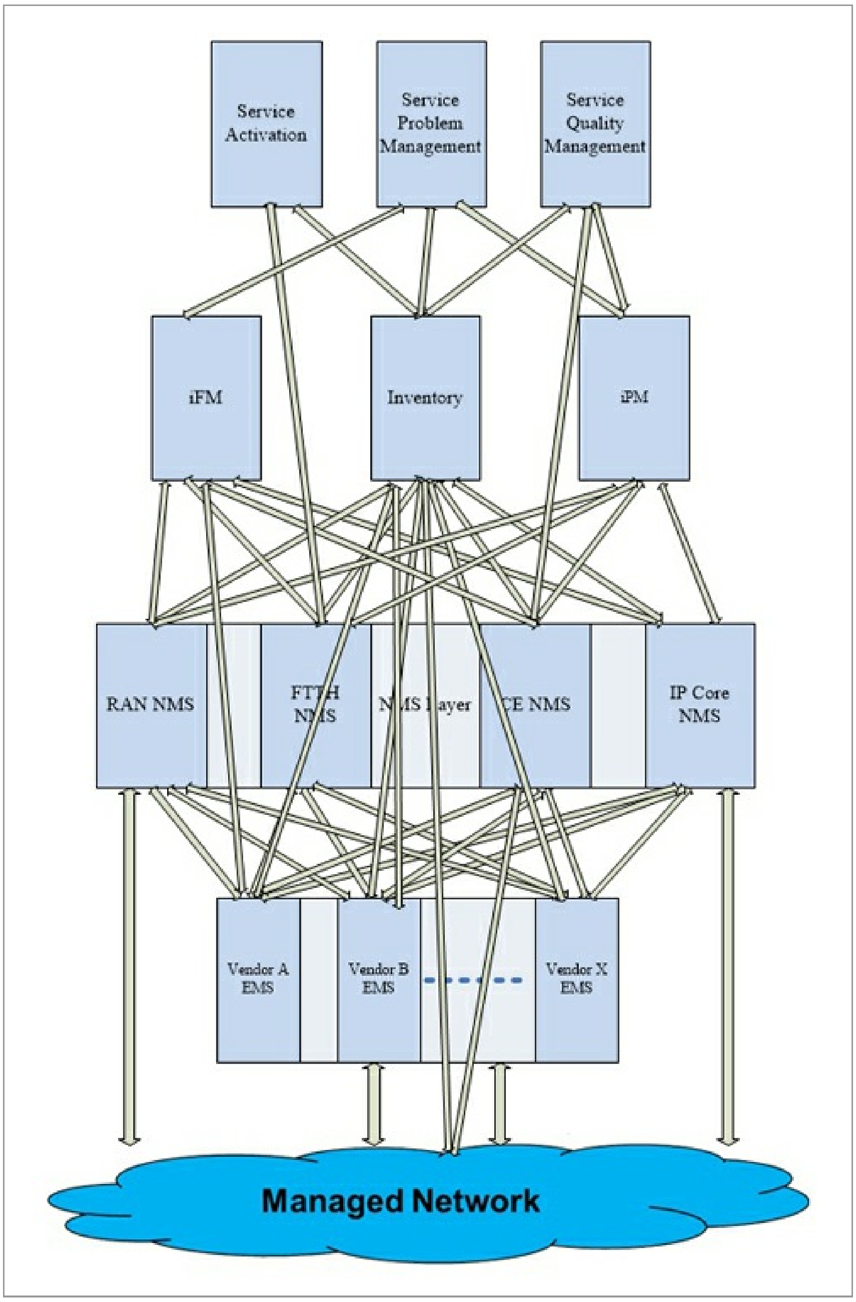

Fig. 1 - Existing OSS Ecosystem

Problems with such existing OSS ecosystems include maintenance complexity, high costs, low operational efficiency, and long service time to market. Furthermore, each time a new system is added, operational complexity and cost increase exponentially, while efficiency drops dramatically. Technically, the information stored in each system either becomes a data island or requires intensive integration and synchronization. It also becomes very expensive and time consuming to ensure data consistency and integrity, and the functionalities of multiple domains becomes harder to implement as the number of systems grow. In one sentence: there are too many problems to describe and the pain lives in every user of every operator that hasn't gone through a modernization exercise.

# Solutions

Existing OSS ecosystems need to be transformed into more streamlined and modernized OSS ecosystems, the key components being:

- Single platforms capable of interacting with the management network at the resource layer
- Single set of interfaces to a unified resource management layer at the service management layer
- Modernized NMS systems that provide a unified set of interfaces to the advanced resource management layer for northbound and the managed network for the southbound (best if standardized)
- Accurate inventory management systems that can boast data consistency and integrity
- Across-the-board OSS functions for all networks
- End-to-end service management layers that manage the entire services lifecycle

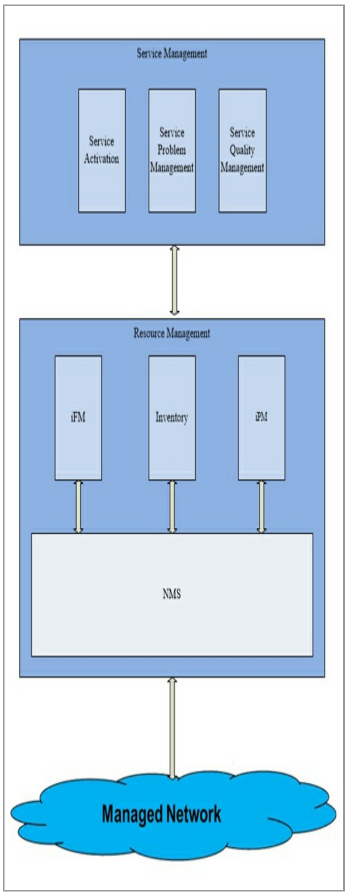

Fig. 2 - Modernized OSS Ecosystem

After OSS transformation, integration can be dramatically streamlined. When networks only need work with a single platform, they are freed from the burden of serving outdated OSS, thus releasing more resources for services provided to end customers. After transformation, functions covering multiple domains can reside either inside a single platform or share the same data from the single platform — refocusing them on solely internal application logics. The fundamental component of such OSS modernization is NMS modernization, because it resides at the bottom layer on which all upper layers depend.

Benefits of a modernization OSS ecosystem include: operation cost savings, reduced maintenance complexity, accelerated service time to market, and agile problem resolution.

Besides the obvious benefits of OSS modernization, there is the added one of eliminating the need for an EMS, except in very rare cases where it might still play a part. In a modernized OSS ecosystem, each layer does what it is supposed to do with single interfaces to the south and north.

# Challenges in the OSS Modernization Process

There obviously will be many challenges to achieving OSS modernization goals. One is to select and match the right software products with the right architecture and capabilities so that scale up and down can be achieved. Products with symmetric distributed architecture should be considered to ensure scalability, performance, and reliability. Also desirable is an NMS that has all FCAPS (Fault, Configuration, Accounting, Performance, and Security) functions at both the resource layer and the service fulfillment and assurance layer.

Another challenge is to ensure the transformation team consists of operator, system integrator and product vendor — all of whom should have a good understanding of OSS at different layers, as well as other important characteristics: high standards of professionalism, rich experiences in delivering complex OSS systems, and an open mind and dedication to ensuring a seamless transformation.

During the OSS transformation process, it is important to avoid service disruption and dramatic change. That means the transformation process will require solid support from top management teams within not only the operator, but also the SI and product vendor, not to mention extensive support of users inside management teams.

# Methodologies for Success

With the large number of existing systems and ever-growing number of network equipment and services involved, it is a fact that the OSS transformation process can be very painful and costly, even it is well planned and implemented. As relevant standards (such as TMN from ITU or the Business Process Framework from TM Forum) separate OSS systems into service and resource layers, OSS transformations should also be done separately at each layer, as previously depicted in Figure 2.

Theoretically, OSS transformation can be done from the bottom up, or top-down, or from all layers at once; however, just like the network protocol stack, the OSS upper layer depends on lower layers, but not the other way around. That means OSS modernization should start from bottom layer. Once the bottom layer is done, then move on to the next layer on top. In other words, the bottom-up approach will reduce the pain in OSS modernization processes and save money in the long run.

The following steps provide an example of how to reduce OSS transformation pain:

**Step 1: OSS Modernization at NMS layer** — Deploy a single NMS that is capable of managing a network without an EMS. The desired NMS should be, without EMS, capable of network and resource auto-discovery, fault and performance data collection, network-element (NE) configuration, and provide northbound interfaces (NBI) for upper OSS application or service management layers. With the deployment of desired NMS, existing EMS can be removed one by one, thus making the deployed NMS a single NMS for the entire network.

**Step 2: OSS Modernization at OSS Layer** — Existing fault management, performance management, customer management, and inventory management systems can be integrated with the desired NMS, or the functionalities can be transformed to reside inside the desired NMS. In the case of integration, southbound interfaces (SBI) of existing OSS layer applications can be migrated to adapt to the NBI of the desired NMS. Obsoleted NBIs of the NMS can then be turned off.

**Step 3: Service Management Layer Transformation** — A single service management system capable of providing all fundamental service management functions and data should be deployed to enable service activation (fulfillment), service problem management, and service quality management (service assurance) for all departments involved in service lifecycles.

# Measuring OSS Modernization Success

How can an operator measure the success of OSS modernization? We can take NMS modernization as an example. Existing OSS ecosystems often have multiple enterprise and carrier-grade NMS systems, along with open source solutions. Almost all of these systems focus only on one area of NMS functionality (such as fault, performance or configuration management). As a result, NOC engineers often have to use many different systems to procure basic information about their networks. With duplicated data in each system, it becomes very difficult for them to know the status of the real network. This makes troubleshooting and root-cause analysis virtually impossible, as NEs are tied up serving multifarious network management systems. Also, network configuration changes are difficult to manage for the same reasons.

In these scenarios, a modernized NMS translates into the ability to use a single software platform to discover networks at layer 2 and 3, providing traditional NMS functions such as inventory, fault, performance and configuration. In addition, other NMS functions required of modern network are provided, such as: packet data protocol (PDP) context management, corporate access point name (APN) management, location-based advertising management, tap aggregation management, configuration change detection, NetFlow management, IP SLA, and data forecasting.

This is when it becomes evident that the modernized NMS has triggered a successful OSS modernization process.

# Further OSS Modernization Points to Consider

In general, successful OSS modernization often depends on products with modern architecture and design. These products need to be designed with OSS transformation and modernization in mind, possessing symmetrically distributed architecture. These products should employ TM Forum's well-defined Business Process Frameworx suites of principles and standards (i.e., eTOM, SID, and TAM).

And because OSS software systems are often very large in scale, the requirements from each operator will be very different in terms of the details, even within the same functional domain. The users of OSS systems from different operators can have very different perspectives, experiences and knowledge. A one-size-fit-all approach, such as those of packaged systems, will no longer be good practice when moving toward OSS modernization.

Component-oriented and product-line software engineering is a good way of meeting the challenges of software solutions development and deployment environments goals. OSS software vendors that own a rich set of modernized OSS software components will eventually win out in the OSS modernization process. These OSS vendors' product components can be either directly assembled or extended easily to solutions so as to meet each customer's needs and requirement. This component-oriented and product-line software engineering approach will ensure each and every operator has its own look and feel in the delivered product and solution, while software quality and agility can be maintained at a low cost.

# Looking Ahead

As we head into the digitally enabled future, NFV and SDN are going to be major forces driving OSS transformation as the existing OSS will no longer work in tomorrow's environments. According to ETSI’s NFV framework, management functions of the NFV network will be carried out by virtual infrastructure manager (VIM), network function virtualization manager (NFVM) and MANO (management and organization) to manage infrastructure for virtual machines, NFV instances and the entire virtual network sector. The resource management layer will need to transform accordingly.

The separation of switching and control logic in SDN will also dramatically affect NE-level management. More importantly, with NFV and SDN, network functions will no longer depend on specific hardware or NEs, as any faulty element will be automatically corrected or replaced by another. For that reason, resource management in OSS and management components in NFV and SDN will require a unified data model such as TM Forum’s Information Framework (SID) and a standard process (such as TM Forum’s Business Process Framework) to facilitate each other. The distributed nature of MANO and SDN control modules will require distributed OSSs to match up. With management functions embedded inside NFV and SDN, component-oriented software can truly provide great benefit to both NFV and SDN.