ONUG ORCHESTRATION AND AUTOMATION WORKING GROUP

ONUG ORCHESTRATION AND AUTOMATION WORKING GROUP MEMBERS

Brian Silverman, *Sr. Architect, Network and Cloud, McKesson*

Carlos Matos, *Executive Director, Cybersecurity and Risk, JP Morgan Chase*

Ted Turner, *ONUG Board Member*

Bruce Pinsky, *Distinguished Engineer, Intuit*

Michael Haugh, *VP, Product Marketing, Gluware*

Kiran Sirupa, *Director, Marketing*

Nikhil Vyakaranam, *Product Management*

Chris Wade, *Co-Founder, CTO, Itential*

Dave Hegenbarth, *VP Systems Engineering,* **Pliant.io**

Mike Kouri, *Product Manager, Megaport*

Rich Martin, *Senior Technical Marketing Engineer, Itential*

Introduction

Most enterprise IT organizations have the goal to implement, increase or improve automating lifecycle management of the network infrastructure to deliver improved agility, compliance, and security to the applications and end-users that rely on it daily.

Automation and orchestration can mean many different things within an organization depending on who you ask, their role, and what they are trying to accomplish. A senior manager may see SD-WAN as a technology that can enable automation, while a NetOps engineer may consider a script or a vendor tool to automate tasks on a network. A DevOps engineer would consider a full-service delivery pipeline to automate the development, testing, and deployment of an application into production.

The majority of enterprise companies have begun to adopt the usage of public cloud infrastructure which has highlighted the use of automation to deploy and change resources and how the traditional network orchestration and automation have fallen behind. A recent Gartner research paper(1) estimates that 70% of network change activities are manually-driven resulting in an estimated 2% that result in some form of error. The paper also estimates that between 10% and 35% of configurations in enterprise networks are unnecessary. This highlights the need for automation to enable agility, reduce outages and troubleshooting time and also increase security.

The purpose of this document is to describe the taxonomy and terminology for orchestration and automation so that enterprise IT organizations can communicate using well-described terms and methods.

Feedback from the ONUG community is that there are a lot of individual automation solutions available and in use in their organizations, but there is a lack of orchestration and integration capabilities to move from task-based to process-based. This document will also look to highlight the areas of need and suggested features and capabilities required when selecting solutions.

Domains

One of the first items to describe is what is being automated, and where does it reside. The broad categories used to describe network domains include:

- **Local-area-network (LAN)**, also known as the "Campus" - these are generally made up of routed and switched infrastructures where devices like PCs, servers, printers, wireless controllers, and more are connected to physical Ethernet ports for access to the network.
- **Wide-area-network (WAN)** - is generally made up of routers, switches, and other transport services that deliver network services to all the locations required by the enterprise. Transport services, like circuits, MPLS, SONET, Carrier Ethernet, and others are typically provided by a telecom carrier. The management of the network devices, like branch routers, backbone routers, and more may be managed by the carrier - referred to as a "managed service", or could be managed directly by the enterprise.
- **Data Center** - is made up of compute, storage, networking, and more. The data center servers are typically where enterprise applications are hosted along with the required storage and network infrastructure required to enable connectivity for those applications out to the end-users.
- **Wireless Network**, also known as "Wifi" - is made up of wireless LAN controllers which control access points that provide the wifi network that PCs, tablets, phones, and other devices use to connect to the network.
- **Cloud** - is a generic term that can be used to describe public resources (compute, storage, and networking) provided by companies like Amazon AWS, Microsoft Azure, Google Cloud, and more. It can also refer to "private cloud" which is built and maintained by an enterprise but leverages similar technologies to the big public cloud providers.
- **Security** - is unique, since it is not a typical domain since there are generally components in all domains, yet often organizations have separate security operations (SecOps) teams who use their own specific tools for O&A to manage firewalls, intrusion detection, proxies and more.

As applications for the network continue to expand, additional domains and/or technologies become part of the landscape including things like:

- **Internet-of-Things (IOT)** - has many applications and adds hundreds to millions of new endpoints to the network for use-cases such as supply chain management and tracking, medical devices, sensors in manufacturing applications, and more.
- **Operational Technology (OT)** - enables automated business operations and is critical for distribution and manufacturing. However, technology often predates modern API and management frameworks and requires rigorous external security controls.
- **Edge Computing** - looks to push compute power closer to the edge of the network (out from the data center or public cloud) to improve response time (latency) and bandwidth. Most of the use-cases are around driving digital services requiring processing power related to the coming faster wireless edge delivered by 5G

Enterprise IT is challenged with managing legacy infrastructure while transforming and enabling new technologies and services. Most organizations are burdened with technical debt caused by older networking equipment, unnecessary configurations, and lack of standardization enforcement. These issues are driving the need for automation to decrease the technical debt and free up IT resources to work on new strategic priorities. Even when new (greenfield) systems are deployed, like SDN technologies including SD-WAN, automation must be considered to minimize manual tasks and processes in service management.

Devices

Devices (physical or virtual) that make up the network infrastructure include:

- Routers
- Switches
- Firewalls
- Load-Balancers
- WAN Optimizers
- Wireless LAN Controllers
- Wireless LAN Access Points

While these devices exist in nearly every enterprise network, they now also exist in the virtual form in the cloud and virtualized environments. Network Function Virtualization (NFV) has had moderate success and organizations must manage these network functions as Virtual Network Functions (VNFs) running in virtualized environments as well as traditional physical devices and appliances. When leveraging a virtualized infrastructure, there will also be a virtual infrastructure manager (VIM) component. Virtualization technologies including hypervisors, like VMware ESXi, and full-stack systems like OpenStack, add virtualization management and can be another component to automate. The deployment of 5G infrastructure is another driver of the usage of virtualized network functions.

Additional network appliances often include:

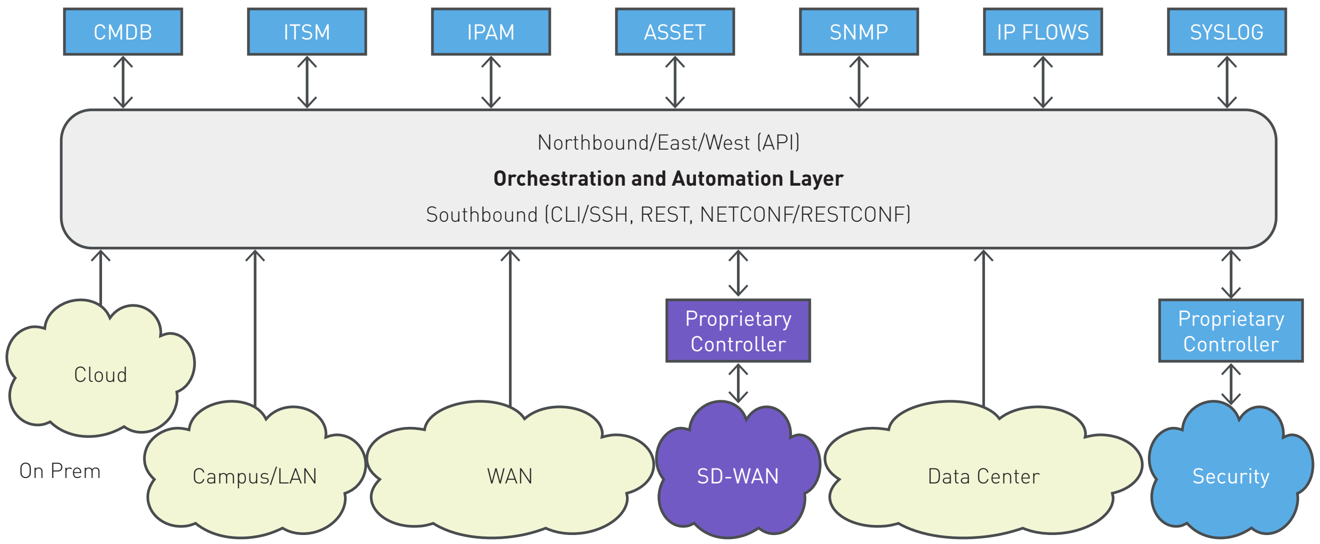
- Domain Name System (DNS) servers
- Network Time Protocol (NTP) servers
- Proxies
- Syslog servers
- Packet brokers
- RADIUS and AAA servers

Management Plane

The network management plane is made up of various tools and services each that provide functions for provisioning, monitoring, and operating the network and services. This is often made up of:

- **Asset Management** - refers to a formal mechanism to store information about all the (networking) assets in the network.
- **Configuration Management Data Base (CMDB)** - is used to store information about networking hardware and software assets. In addition, the CMDB typically stores the details of the assets, like software release versions and configuration back-ups and/or standards.
- **IP Address Managers (IPAM)** - are used to plan, manage, and assign IP addresses used by the network devices. It also can allocate pools of addresses used for DHCP services.
- **IP Flow monitoring** - provides a monitoring infrastructure using a protocol like NetFlow or SFlow to collect information about the IP traffic flows, provide analysis and reporting of the traffic types on the network.
- **Application Flow monitoring** - CNCF (Cloud NativeComputing Foundation) sandbox project is similar to IP flow monitoring but originates with agents injecting application flow information at multiple tiers in the web, app, and database. Applications can be threaded together in loosely coupled groups, and one can map outcalls across one application stack into another application stack. Cloud vendors are now starting to introduce the capability to incorporate Application Flow monitoring in their cloud-based offerings along with other metrics on this list. https://opentelemetry.io/
- **IT Service Managers (ITSM)** - are platforms (like ServiceNow, Remedy, and others) that provide structure and processes to plan, deliver, and operate IT systems.
- **Simple Network Management Protocol (SNMP) based monitoring** - is an internet standard protocol for collecting information using a management information base (MIB) structures that describe the system status and its configuration.
- **Telemetry** - is a new form of real-time monitoring to collect and analyze network information (typically, provided by gRPC, an agent-based protocol) streaming from a network device that has been subscribed to. https://grpc.io/

EXAMPLE REFERENCE OF NETWORK, O&A AND MANAGEMENT PLANE LAYERS



An issue to highlight most enterprise IT faces is establishing and maintaining a "Source of Truth" (SoT). Network administers may believe the source of truth is in a CMDB system, while Network Operations may believe it is on the running network devices. Enterprises must decide how they synchronize information from devices running in the network and offline databases to reconcile the data. This synchronization can often be automated via API integration. Large organizations also may have distributed SoT systems which are federated. In the O&A workgroup, this has been a topic of interest since there are challenges within organizations creating, keeping, and maintaining accurate SoT. In most cases, there are many different sources for things like inventory, IP address, operating system tracking, and more. Most of these systems do not currently have an automated process to reconcile with what is actually in the network.

Users of O&A

The stakeholders of O&A usually include budget holders, technical decision-makers, and operators. These roles include

Executive Management (Director, CIO, CTO) - These are the budget holders who understand the strategic direction of the business and how the underlying technical resources are going to be funded to drive those.

Network Architects - are responsible for the network architecture and typically own network design and standardization. They have deep protocol expertise across multi-platform and multi-vendor. They must consider the lifecycle of a product or solution going into the network.

Network Engineers - are the bridge to operations and oversee implementation of ongoing changes in a network. They typically have a deep level of vendor specialization along with protocol expertise and see the big picture of the network and how the applications are using the network.

Network Operations - are responsible for day-to-day operations including implementing moves, adds, and changes along with the required troubleshooting. They have vendor and platform experience from beginner up to advanced depending on their seniority.

As organizations formally embrace automation additional job roles can include:

Director of Automation - a formal owner of the automation infrastructure, tools, and processes the organization uses and maintains.

Tools Manager - is largely responsible for managing management software provided by vendors and may also manage the development of in-house software and/or scripts

Developer - if an organization is building their own software platform and/or creating a "manager of managers" to automate integration they will be staffing developers with core programming skills

Full Stack Engineer - this role typically defines a staff member who has network engineering skill sets as well as programming or scripting skills and can develop automation for tasks or processes the organization requires.

DevOps Engineer - this role is often associated with the automation pipeline for development and delivery of applications, however, as enterprises look to automate network infrastructure there is cross-pollination to leverage the skillset and tools used in DevOps.

SecOps Engineer - this role will leverage data provided from internal and external sources and build automation in support of organizational security policies and practices that may include a range of actions.

Site Reliability Engineer - is a newer role in an organization includes the design and operational elements including security, performance, scale, and availability. Google has literally written the book on it **https://landing.google.com/sre/books/**

Types of O&A

Network automation, network orchestration, and network programmability are often used interchangeably. Automation networking devices started more than 30 years ago and have their roots in script development in various languages including TCL, Expect, Erlang, Java, JavaScript, and the most popular now Python. Many commercial tools and home-grown approaches are built using those technologies. Network O&A has been evolving and progressing from treating configuration as blocks of text to blocks of code. This transformation generally means leveraging some form of data-modeling to move from low-level CLI commands to more of a programmable format. Some vendors are beginning to directly support data modeling through NETCONF and YANG based data models. Network automation generally has a 'southbound' direct to specific network devices approach and is mostly task-based. Network orchestration generally must be able to speak southbound to many different devices and device types, but also must be able to communicate northbound to coordinate other systems in the management plane.

Network automation — generally refers to automating interactions with network devices (routers, switches, firewalls, load-balancers, WAN Optimizers, Wireless LAN controllers, and more). The majority of the legacy network devices do not have a programmatic interface, therefore they require a command-line interface (CLI) based automation approach to emulate a user. This has been historically challenging since each vendor has its own semantic and the human-readable CLI must be interpreted by the automation system. As vendors modernize their platforms, they are providing programmatic interfaces, such as APIs to automate.

Network orchestration - generally refers to automating interactions across multiple types of devices, domains, and even potentially other related systems in the management plane. Orchestration typically requires the ability to interact with many device types and vendors, potentially across multiple domains and management systems requiring programmatic interfaces including Restful APIs.

Some well-known approaches to network automation include the following terms:

Network Configuration and Change Management (NCCM) - These systems have the capability to capture and maintain an inventory of the network devices, provide a comparison when configs change (drift), some provide configuration audit, automate configuration management, operating system upgrades/patches, the rollout of a new configuration. NCCM tools started coming to the market more than 20 years ago and vary quite a bit in terms of the underlying intelligence they have about the network and the ability to pre-check, post-check, and validate the configuration and operational state. A challenge with this older approach is that these systems generally treat configuration information as blocks of text, not code, so there is a lack of or limited version control and change control when variables change - like values for protocols SNMP, DNS, ACL or more change.

Policy-Based Automation (PBA) - provides an abstraction away from individual device configuration and enables centralized control to device a "policy" which is made up of specific configuration parameters for a device type or role. Many of the technologies that use "software-defined" labeling provide policy-based automation to support implementations such as SD-WAN and SD-Access, SD-DC, and more. This approach is an improvement over NCCM, but often the policies are defined in a template format and template management can introduce new challenges. When performing updates, the PBA systems generally push the full-updated templates to the networking devices, which could be disruptive.

Software-Defined Networking (SDN) - originally was intended to separate the control-plane and data-plane to enable lower-cost networking layer devices that had programmable forwarding tables (like via the OpenFlow protocol). While this approach did not have wide adoption, other SDN technologies have had a lot of success including SD-WAN and SDN technologies in the data center. SDN introduces a Controller layer that provides management functions such as provisioning, monitoring, and configuration management. They also provide a REST-based programmable interface for automation and interworking with other management systems.

Intent-Based Networking Systems (IBNS) - these systems are considered the most advanced from a technology standpoint since the objective can be abstracted up to a business level intent and then the system can derive the required configuration, implement on the network and verify the network remains in that intended state. While this is one of the most advanced approaches, it can involve more time to ensure the design and required features are built properly and ready to consume an intended state to be translated down into the actual network configuration.

DevOps Frameworks - are often referred to as a toolchain that enables a "pipeline" of automated steps to deliver continuous integration / continuous delivery (CI/CD). These often require a high degree of programming proficiency to abstract and automate the compute, storage, and networking required for continual application development and deployment.

A challenge we are observing in this working group is for Network Operations to progress from legacy NCCM based technologies to a more intelligent, scalable, and reliable approach. Newer policy-based, and IBNS systems enable the ability to perform pre-checks, post-checks, and verifications instead of blasting CLI commands at network devices with limited intelligence. Enterprise IT is looking to treat network features as code instead of blocks of CLI and be enabled with a programmatic approach to managing their infrastructure reliably and at scale.

Methods for O&A

- **Script-driven** - Scripting languages like Python, Pearl, TLC, JavaScript, and others are used to write task-based fairly low-level commands to interact with devices to push configuration changes or extract configuration or state information from network devices. Scripts are often run directly from the author's computer. Popular script-driven frameworks are available, like Ansible and commercial frameworks also often provide user-interface which manages the scripts.
- **Model-driven** - Refers to automation platforms that leverage a data-model like JSON, XML, YANG or others which provide an abstracted structure to store and use the data associated with the network devices, configuration variables, and operational state. Leveraging data-models can help to transform low-level command-line-interface (CLI) based devices to a more programmatic mechanism for automation. It can also help to enable automation across multiple vendor device types and platforms by enabling abstraction away from vendor-specific semantics. These platforms must be able to read and write to/from the native CLI and semantics when interacting with the devices.
- **SDN based** - The original definition of Software-Defined Networking called for the separation of the data-plane and the control plane and protocols like OpenFlow was developed to "program" the forwarding table of the network devices which were all centrally controlled from a controller and software applications. While OpenFlow did not have a lot of traction, technologies like vendor specific SD-WAN and SDDC have emerged that provide software control through a controller and applications which simplify the management of the network devices and services. Theses systems generally deliver some form of automation, but are often template-based for configuration and require programming via API to automate interaction with external systems.

AUTOMATION COMPARISON

AUTOMATION	METHOD	OUTCOME
NCCM Based Automation	script-driven	task-based, limited automation, not highly reliable
Policy-Based Automation	data-model driven	template-based, config automation and integrated management, most are integrated closed vendor-specific solutions
SDN Based Automation	data-model driven	template-based, config automation and integrated management, greenfield, specific domains, like SD-WAN
Intent-Based Automation	data-model driven	abstraction available, more reliable due to pre-checks, post-checks, validation, most advanced option but could be longer implementation, outcome driven, can generate required configuration from intent, most options are not for brownfield network

Interworking Systems

A critical capability of an O&A platform is to interwork with other systems in the management plane. With the availability of standards-based API protocols, most products and services used in the network management plane provide a published API to enable programmatic interaction. Enabling and leveraging this capability will be required to orchestrate processes that require interaction with multiple systems, like IT Service Managers, IP address managers, configuration management platforms, and more. An API is an Application Programming Interface, which by definition will require some programming to leverage. This is a key area to understand for Enterprise IT when selecting components to use in their management plane. At this point, published APIs should be a requirement. Systems without an API should be considered technical debt and plans should be made to move away from them over time. Without an API, costly manual effort will be required which also introduces the potential for manual errors. Leveraging these APIs and focusing resources that have programming skills to use these and integrate systems.

Example Use-Cases

Read-Only

- Inventory - Create and maintain a network source of truth (SoT), starting with an authoritative network object inventory
- Config change monitoring - Monitoring and reporting on config changes (drift)
- Config audit - often driven by regulatory and compliance, but also useful for company policy ad-hoc audits and security policy
- Troubleshooting/state capture/data gathering

Read/Write

- Operating system patch/update
- Configuration management
- Network security policy management
- Zero-touch provisioning (deployment)
- Service provisioning
- Auto-Remediation

Analytics

- Understand data patterns and network utilization
- Provide predictive capability including 'what if' before a change

This list is just a few examples of use-cases. The O&A workgroup will also be publishing more detailed examples.

Best Practices

Perform a detailed assessment - The assessment should be pretty comprehensive of people, processes, and technology used within network engineering and operations.

Align with a strategic project/corporate initiative - Getting 'buy-in' within your organization will be critical to success. Often times, there are strategic programs that are started, like 'operational excellence' an automation project can be tied to achieve cost-efficiency through collaboration

Automate the read-only first - Enterprise IT is inherently risk-averse since most are benchmarked on uptime and availability of the network. Everyone has a story of how a network automation attempt in the past caused some degree of a network outage. To lower that risk and gain confidence in the system, prioritize low-risk use-cases first, like read-only items to gain insight and plan an automated change next.

Compliance - Most organizations have requirements to be in compliance with internal or external specifications. Identifying an automation use-case to help audit or achieve and remain compliant can add a lot of value to the organization.

Get the quick win - Look for opportunities to get quick wins with automation and show its value. This could be automating to reduce troubleshooting time, or automating a regular task, like password rotation.

Establish a process to test, verify, and then deploy - Since automation can introduce risk, it is critical to have well-defined processes to test, verify, and the document before the actual deployment in production.

Pre-check, post-check, verify, rollback - Ensure the automated system is performing a pre-check, post-check, and has verification of the change. If a change is unsuccessful, it must be able to be rolled back.

Automate within domains, then across domains - Often, automating within a domain, like VLAN changes in the LAN or ACLs in the WAN offer a good starting point. Next, an organization should look to automate end-to-end processes and reduce all manual steps.

Establish Well Defined Processes to Automate

Most Enterprise IT organizations have a number of well-defined and documented processes for lifecycle management related to Network Operations. These are often related to Move, Add, Change, or Delete (MACD) activities of which the acronym comes out of the telecom industry. An issue that has been highlighted in the O&A working group is that when it comes to many of the activities performed in Network Operations, many are not well documented, 'tribal knowledge' activities of which a limited sub-set of individuals are able to execute.

Establishing documentation for each task and process will help to enable automation which will enable repeatable, reliable changes available to all of the operations engineers.

Software like Microsoft Visio or Lucid Chart can help to create and publish these processes.

Some standards are emerging to help provide structure to how these business processes can be defined and accelerate the implementation in O&A platforms. One example is the Business Process Model and Notation. **http://www.bpmn.org/**

Where to Start

Some questions to start with:

- What do I need to automate/orchestrate?
- What should I do first?
- What do my teams spend most of their time working on?
- What network domains are the highest priority?
- Who are the teams involved in the process?
- (For each initiative) what is the return-on-investment (ROI)?
- How can the results of network automation be measured? Are there SLAs or other metrics that can be used for before and after?
- What tools, platforms, products are currently used? What new ones should be considered?
- Is the need for automation/orchestration *strategic* or *tactical*?

Questions to help identify Network Automation use cases to start with:

- Is there a task/process that has a large backlog of requests?
- How long does this specific task take for a human to complete?
- How often is this task run?
- How many devices does this task need to run against?
- If the task must run in a scheduled time (maintenance window), how many times can it be completed inside the window?

Recommendations for an Orchestration and Automation Solution

These capabilities can be provided by more than one solution working together to accomplish the business goals.

System

- **Multi-vendor/Multi-platform** - the solution must be able to provide O&A for the vendor products used in the enterprise network which is often multi-vendor and made up of multiple platforms (routers, switches, load balancers, firewalls, wireless LAN controllers, WAN Optimizers and more)
- **Authentication with Role-based-access** - The solution must enable the definition of users and their rights within the system. Often these integrate with RADIUS, LDAP, TACACS, and other existing systems
- **Redundancy/Failover** - the system must provide a mechanism to back up the data used by the system and failover to a secondary system upon failure.
- **Database integration** - the system must provide an integrated (or integration with a) database to store the data used for O&A activities.
- **Logging** - the system must provide comprehensive logging to determine what happened for purposes of audit or troubleshooting.
- **Scheduling** - the system must support the ability to schedule and automatically run O&A events

Device Management

- **Inventory** - the system must provide the ability to pull the device inventory information either directly (SSH/Telnet) and/or through API. Inventory typically consists of vendor, model, operating system, serial number, and more information.
- **Capture state information** - the system must be able to provide the ability to capture state information from the managed devices to provide pre/post verification checks as well as potential troubleshooting. The state information can include items like interface state (up/down), protocol state, route tables, route counts, ARP tables, and more.
- **Capture configuration** - the system must be able to capture/snapshot the current running configuration on the devices under management (directly or via API). This is generally a requirement for brownfield network devices to ensure they can be recovered if an incorrect change creates a service disruption.

Integration

- **API/Programmatic interface** - O&A systems must provide the ability to programmatically interact with external systems and also be interacted with directly. This is typically provided by a published API.

Summary

As the O&A working group gets started, the logical starting place was to define the taxonomy and terminology used. In parallel, many use-cases are being defined with example approaches as to how to apply orchestration and automation solutions. Several challenges have been identified and are being discussed including:

- Many disjointed automation solutions with a lack of integration and orchestration
- The challenge of having many systems in the management plane to integrate and the requirement of published APIs
- The issue of creating and maintaining (one or more) Sources of Truth (SoT) for the network infrastructure
- The addition of new domains including cloud, IoT, and 5G to name a few
- The challenge of technical debt and legacy automation approaches which have limited progression
- The integration of new systems and approaches including SDN, SD-WAN, policy-based, and intent-based networking
- The need to build and define a business case along with a return-on-investment (ROI) to help justify investing in O&A
- Changes needed in the mindsets and skill sets of teams designing, building, and operating network infrastructures

As this working group progresses, the items above will help to drive work and deliverables to help enterprises accelerate O&A projects within their organizations.

Additional Terminology and Acronyms

API - Application Programming Interface, often described with the method, like a RESTful API (using REST calls). Published APIs are used to interwork systems in the management plane and automate modern network systems which provide an API interface.

Automation - Specific to the network space, this is using a programmatic method to replace a manual task or process related to operating and configuring the network devices and/or services that run on the network.

Abstraction - Specific to the network space, this is the concept of modeling a higher level construct, usually in a data-model, moving away from lower-level vendor-specific CLI and semantics

Cloud management platform (CMP) - Is a platform, usually specific to the cloud provider, that provides management of the servers, storage, and networking resources. These usually offer graphical user interfaces as well as programmatic interfaces (API) to perform management functions.

CI/CD - continuous integration/continuous deployment - this refers to the process of combining continuous integration to merge and share developers' contributions to a published version as often as several times a day. Continuous deployment refers to the process of automating the deployment.

Command-line interface (CLI) - is a low level, usually vendor-specific, human-readable command set to configure and execute operational commands to display operational state from a network device.

DevOps - is a set of practices that combines software development and operations to shorten the system development cycle. While this is generally a concept related to the development and deployment of software, the automated pipeline approach is a model often referenced where network operations are working towards.

Provisioning - in networking refers to the process of preparing network equipment to provide a new service which involves applying the configuration

SLA - Service level agreement, usually associated with a measurable metric of performance, uptime, and availability of the network

References

1 - Gartner, Nov 2019 "How to Reduce Techincal Debit in Enterprise IT"

ADDENDUM

Draft O&A - Use Cases

Network Automation and Orchestration deliver tangible business outcomes. However, organizations are encouraged to adopt an incremental approach to introducing Network Automation. The goal should be to start small, achieve quick wins, gain confidence from the executive management to expand further. In that context, the O&A workgroup has identified a few high impact use-cases that can show immediate ROI to the business by reducing operational complexity, eliminating the manual processes, and accelerating project delivery.

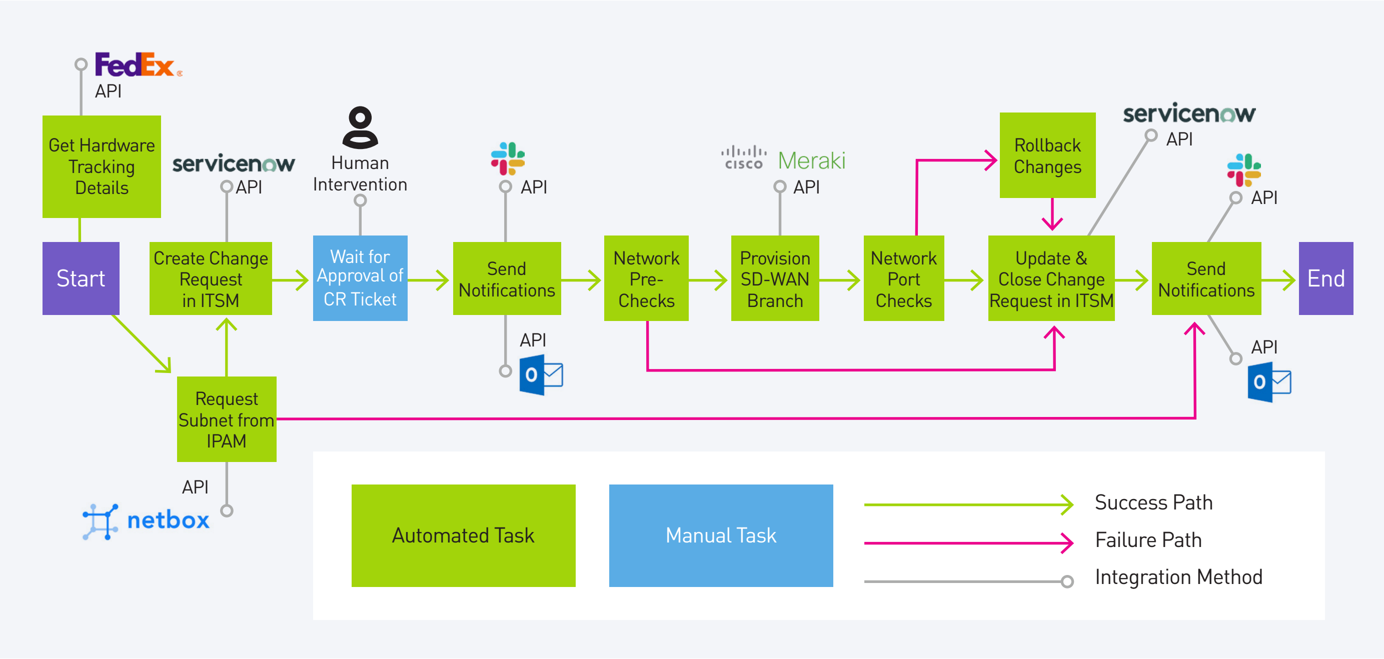
Enterprise Branch Onboarding:

Customers struggle with how to make SD-WAN branch Onboarding a Zero Touch Process. Manually moving between systems and documenting the change occurring can take days and weeks. The systems involved are API accessible and therefore can be automated as well as orchestrated.

Sample Onboarding Task List:

- Get location details
- Track device shipment (Shipping Carrier)
- Notify delivery
  - o Update Change Request (Change System)
- Get IP (IPAM)
  - o Update Change Request (Change System)
- Open Change Request (ITSM System)
- Send Notifications (Email & Team Collaboration)
- Identify Network or Create Network (SD-WAN Controller)
  - o Update Change Request (Change System)
- Pre Checks - Device not in use, IP not in use, etc.
  - o Update Change Request (Change System)
- Claim Device against Network (SD-WAN Controller)
- Apply configuration to Device Profile (SD-WAN Controller)
- Fetch activation key
- Perform Post Checks
- Update monitoring (Monitoring System)
- Update and Close Change Request (ITSM System)
- Send Notifications (Email & Team Collaboration)

Example Diagram of Multi-Domain SD-WAN Branch Automation:

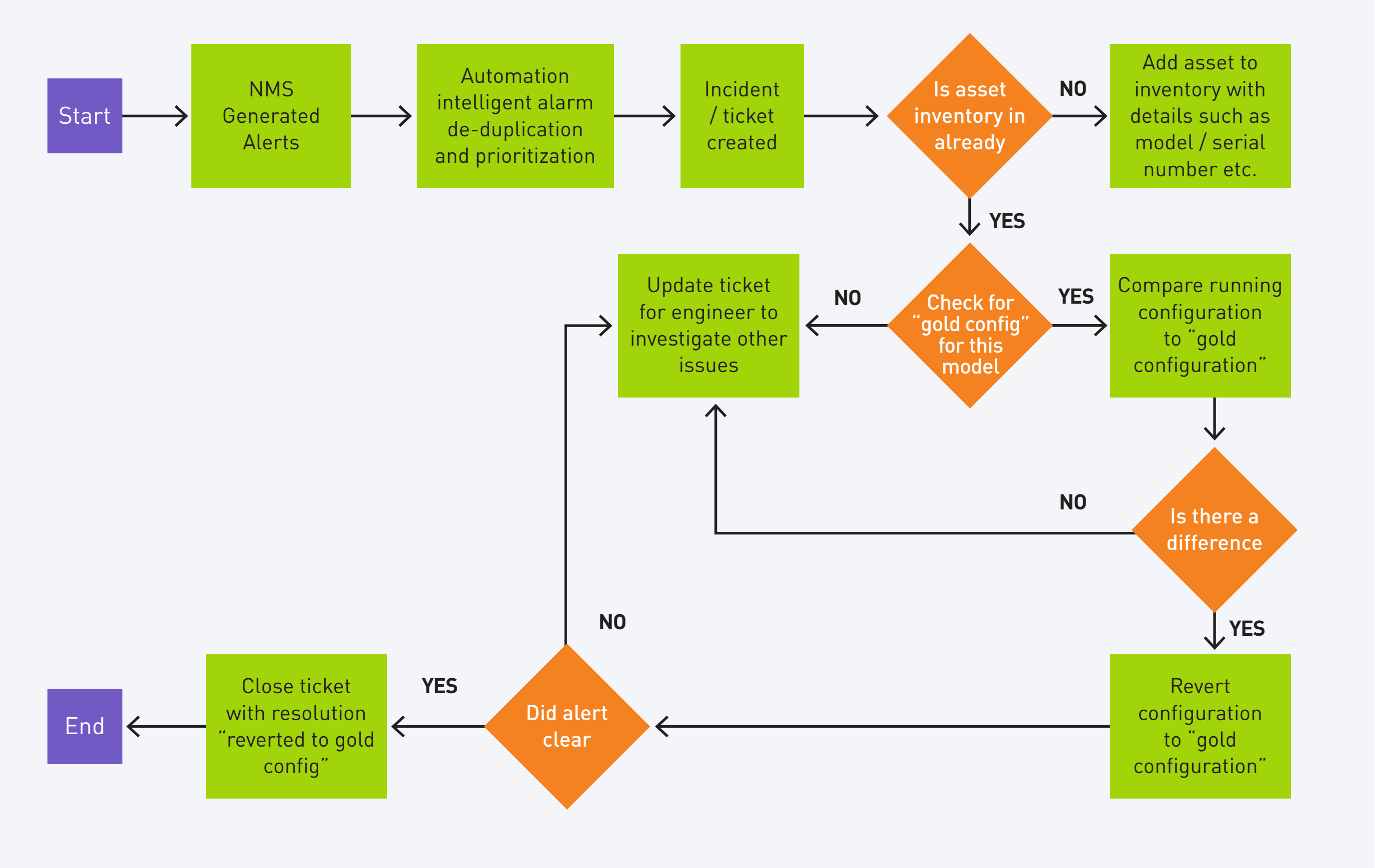


ITSM Lifecycle Automation:

Using a service desk management tool such as ServiceNow or ConnectWise we can use automation to help document and remediate issues within the network environment. A set of workflows could provide the following logic:

- Take inbound alerts via SNMP traps or a message queue bus such as Rabbit MQ - deduplicating alerts if necessary
- Open a trouble ticket in the ticketing system
- Compare the running configuration of the device at issue with the Golden configuration - create a list of missing / extra commands necessary to make the device compliant
- Configuration delta applied
    - o Automatically open a change control ticket with the recommended changes and allow the ITSM system approvers to approve and then schedule the configuration change
    - o Directly apply configuration changes to the device
- Check in the new running device configuration to a code repository such as GitHub
- Close the change ticket
- Post-change checks
    - o Ensure that the alarm from the NMS has cleared and close the trouble ticket
    - o Move the trouble ticket to Tier 3 support if configuration change does not resolve the issue

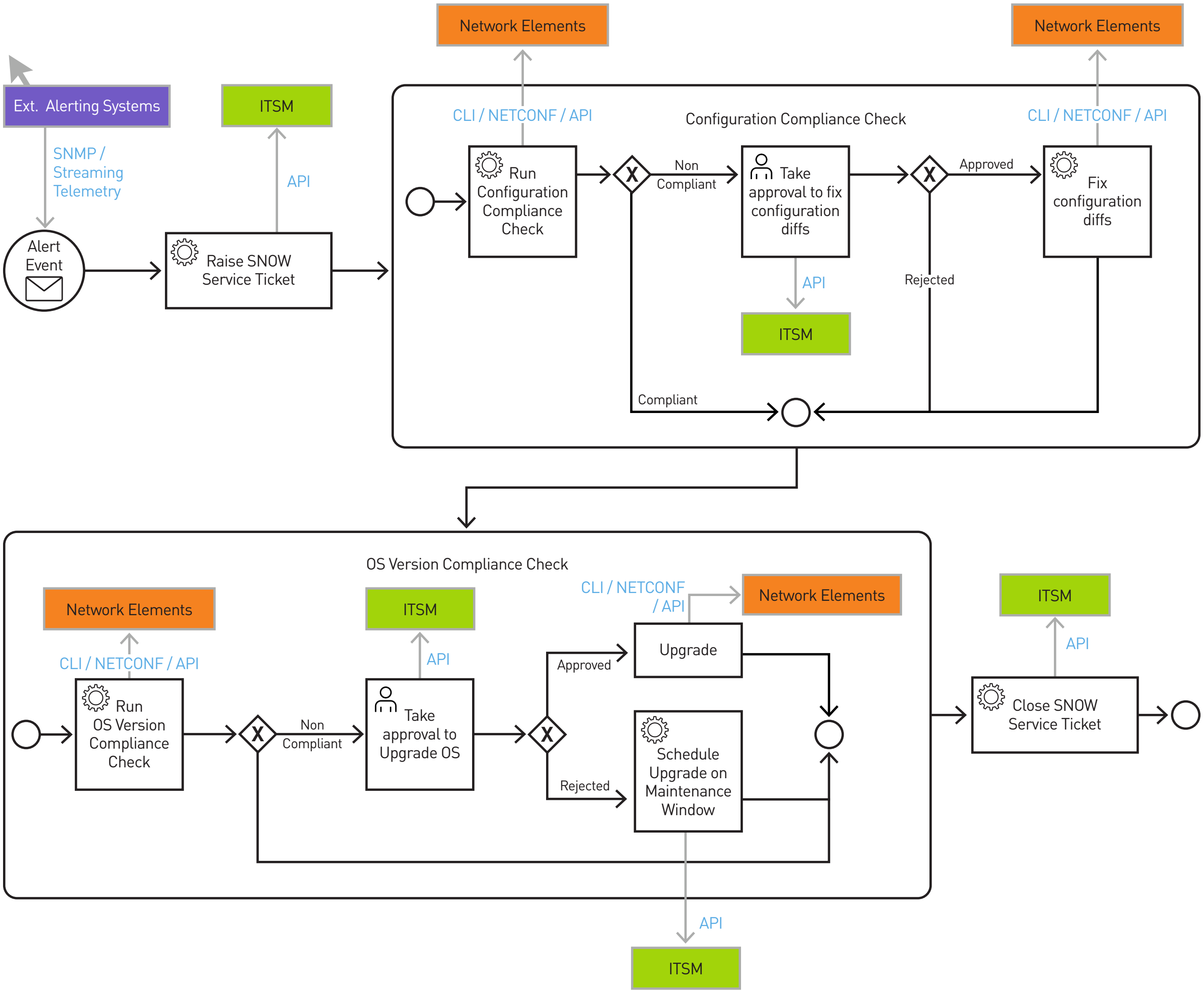
Example workflow diagram:


Start
NMS Generated Alerts
Automation intelligent alarm de-duplication and prioritization
Incident / ticket created
Is asset inventory in already
NO
Add asset to inventory with details such as model / serial number etc.
YES
Update ticket for engineer to investigate other issues
NO
Check for "gold config" for this model
YES
Compare running configuration to "gold configuration"
NO
Is there a difference
YES
Revert configuration to "gold configuration"
NO
Did alert clear
YES
Close ticket with resolution "reverted to gold config"
End

Automated Network Troubleshooting

Network troubleshooting is sometimes complicated and requires data gathering and analysis from many different tools. Known troubleshooting scenarios could be quickly identified and rectified with automation. Below is an example of BGP Flap remediation.

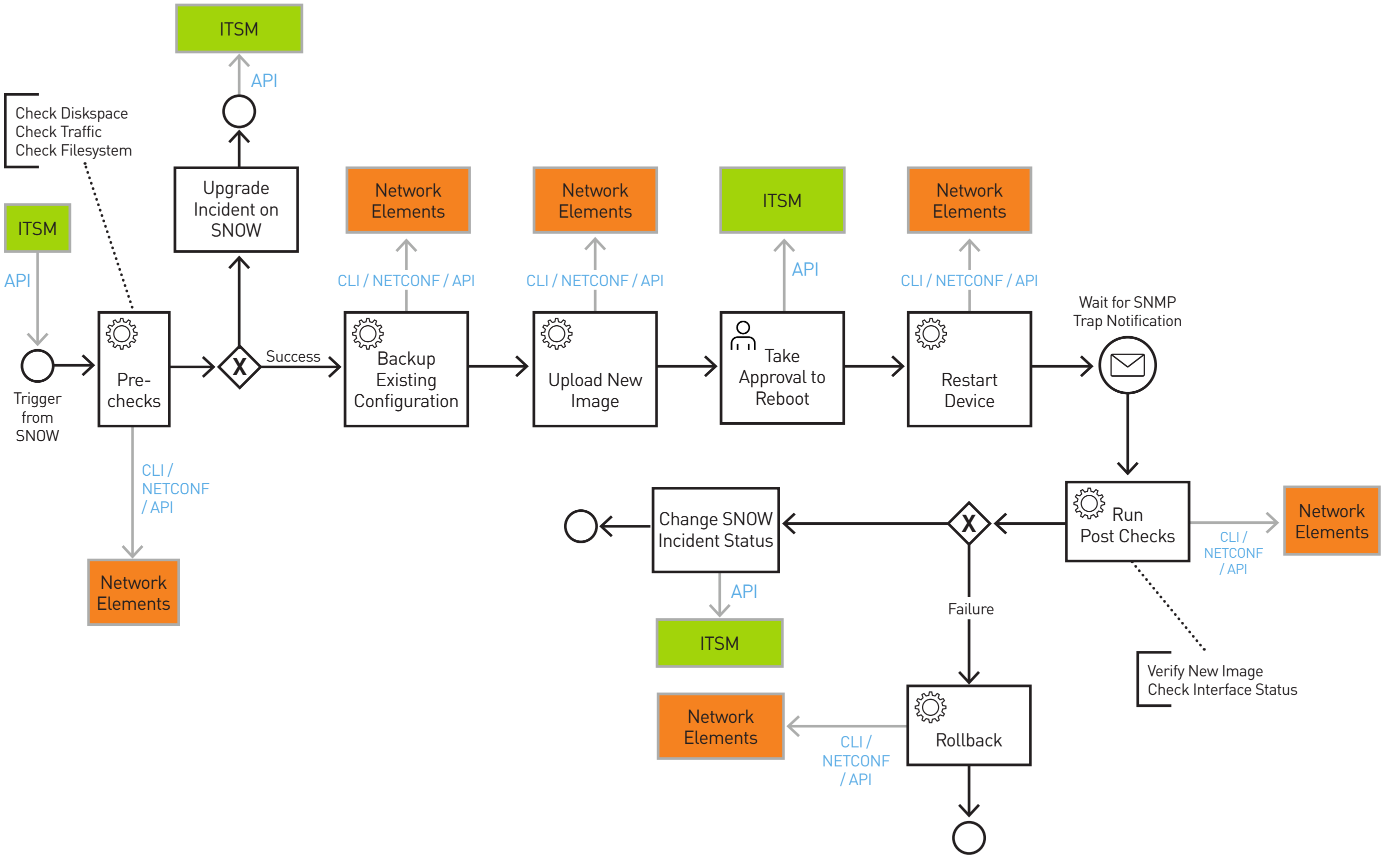
- Receive critical alert due to excessive BGP flaps on a particular device
- Raise an incident ticket on Service Now
- Run predefined compliance checks on the device. Compliance checks can yield configuration and OS issues.
- Compare with configuration compliance policy for the device and automatically apply fixes to eliminate configuration diffs.
- Schedule OS upgrade issues for the maintenance window
- Rerun compliance checks to validate device
- Close incident ticket on Service Now



Automated Software Upgrades

OS upgrade is an activity that can quickly gather complications, confusion, and delays as it rolls forward. An upgrade is a complex process involving prechecks, post checks and approvals which are mostly manual in today's processes. Automation can simplify the method-of-procedures and remove human errors

- Raise upgrade ticket on service now that triggers OS upgrade automation workflow
- Run Pre-Checks
    - o Check if enough disk space is available
    - o Check if traffic is running on the interfaces
    - o Check file system
- If prechecks succeed go to step 4, else change incident status on ServiceNow
- Take configuration backup
- Upload new image
- Ask for approval before applying new image and rebooting the device
- Restart device on approval
- Wait for SNMP trap to know when the device is back online
- Run post-checks
    - o Verify new image is applied
    - o Check interface status
- If post checks succeed go to step 11, else rollback to an earlier image
- Change status of the incident in service now



Appendix - A. Reference notation potentials:

Gherkin / Cucumber

- https://en.wikipedia.org/wiki/Cucumber_(software)#Gherkin_language
- https://cucumber.netlify.app/docs/gherkin/reference/

Business Process Modeling Notation (BPMN)

- https://en.wikipedia.org/wiki/Business_Process_Model_and_Notation

POWERING THE NEXT GENERATION
OF DIGITAL ENTERPRISES. TOGETHER.

ONUG
PO Box 702
Hingham, MA 02043

Produced in the United States of America

Editor Charlene O'Hanlon
Cover design and interior by Giulia Fini

For further information about ONUG, these and other publications, special discounts for group ticket purchases to attend ONUG conference, or for information on booking authors for an event, please visit our website at onug.net