ONAP
OPEN NETWORK AUTOMATION PLATFORM

Bell Canada

Case Study

Bell automates a significant amount of manual configuration, recovery, and provision work by using ONAP in production across multiple use cases

Bell

COMPANY:

- Canada's largest communications company
- More than 22 million consumer, business and wholesale customer connections

CHALLENGE:

- To enhance customer experience through rapid service provisioning
- Streamline operations through automation

SOLUTION:

- Linux Foundation's Open Network Automation Platform (ONAP) project
- Internal open source development and support team

BENEFITS:

- Automation implemented by numerous business units
- Manual recurring input reduced

"Automation can translate to a competitive advantage."

–OLIVIER PHENIX, SENIOR MANAGER, NETWORK ORCHESTRATION, BELL CANADA

The Business

Canada's largest communications company with more than 22 million consumer and business connections, Bell provides advanced broadband wireless, TV, Internet and business communication services throughout the country.

Bell Media is Canada's premier multimedia company with leading assets in television, radio, out of home and digital media. Founded in Montréal in 1880, Bell is wholly owned by BCE Inc.

Challenges

Before embarking on the path to NFV/SDN, services were designed using physical network functions (PNFs). The PNFs, in turn, were plugged into physical networks.

To provision a new service, Bell would send technicians into the field to install and configure the equipment. In fact, different technicians for Layer2 vs. Layer3 configuration would often result in multiple truck rolls. In addition, once the service was operational, Bell's strategy to recover from an outage was the existing L1/L2/L3 model where the support staff would take calls, analyze problems, and proceed with the recovery steps. Clearly, this approach was largely manual, where a support person would debug the issue, run diagnostics, review the logs, and develop a resolution strategy.

Realizing that it was taking too long to provision services and correct problems, the Bell team knew that automation would be central to providing a better customer experience with the ever-increasing demand for new services.

Even though the need for full automation was several years away, the Bell team recognized in 2016 that it would take a number of years to put together a mature automation solution and created a Network Service Orchestrator (NSO) project with a sense of urgency and determination.

Solution

The Bell NSO team started its journey by evaluating proprietary solutions using the standard procurement approach of issuing requests for proposals.

In 2016, they were unable to find a solution mature enough to meet their current needs. When AT&T open sourced OpenECOMP in late 2016 (which subsequently merged with another open source project called Open-O to form ONAP), the Bell team tried it out and were pleased with the results.

ONAP is a comprehensive platform for orchestration, management, and automation of network and edge computing services for network operators, cloud providers, and enterprises. Real-time, policy-driven orchestration and automation of physical and virtual network functions enables rapid automation of new services and complete lifecycle management critical for 5G and next-generation networks. (source: onap.org)

At that time, several individual departments also realized that automation was a must and started experimenting with their own solutions. However, the NSO team recognized that a flexible platform such as ONAP would standardize operations to avoid work duplication.

Since then, ONAP has been foundational for Bell's NSO solution. Instead of using the entire project, one of the interesting architectural choices the NSO team made was to use only a subset of ONAP projects.

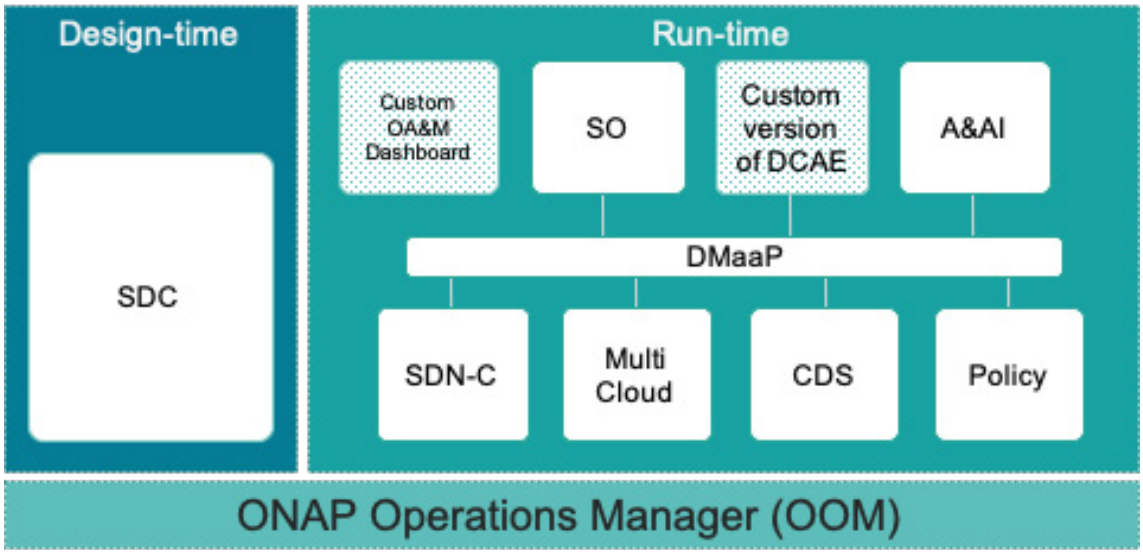

Figure 1: ONAP projects used by Bell NSO Team

Currently, the Bell NSO team uses the following projects:

- SDC: Service Design and Creation for onboarding xNFs and creating service/closed-loop design
- SO: Service Orchestrator for service, NFV and SDN orchestration
- CDS: Controller Design Studio for xNF lifecycle management and configuration
- A&AI: Active and Available Inventory for inventory services
- Policy: Policies for closed loops and other purposes
- Modified DCAE: A custom, in-house runtime that respects both the DCAE architecture and data model
- Custom OA&M dashboard: A custom operational, administration and management portal created by Bell and used in-lieu of VID

The approach of using only the modules the NSO team needed provided focus, allowed time to build expertise, and simplified maintenance.

1 A virtual, physical and cloud-native network function (VNF, PNF, CNF respectively) are collectively referred to as xNF in this document

Open Source Mindset

Moving to an open source project for such a vital piece of functionality required a major mindset change. To transform to this mindset, the NSO team made several cultural changes:

- Bell created a 25-member in-house team to work on ONAP, with support from other team members that build and automate services using ONAP.
- The NSO team adopted an agile methodology with continuous integration/continuous delivery (CI/CD). The team built a CI/CD pipeline with automated testing and engineers were engaged from the beginning of the development process to the deployment of ONAP software.
- Instead of a classic L1/L2/L3 internal support model for ONAP, Bell used a Site (or Systems) Reliability Engineering (SRE) approach to build automation so that the issue gets addressed automatically. The support is provided by software engineers that focus on making the software fault tolerant rather than attempting to eliminate faults.

These changes allowed Bell to put the ONAP Amsterdam release into production in 2017.

The NSO team also chose not to fork any ONAP component. The goal is to stay as close to upstream as possible, making it easier to contribute to and benefit from the upstream community. There is also no need to backport custom patches and Bell upstreams every feature they develop. In fact, the NSO team is credited with several major innovations in ONAP including the ONAP Operations Manager (OOM) project that deploys a containerized version of ONAP onto Kubernetes and the CDS controller.

Bell is currently exploring new opportunities to verify VNF interoperability with ONAP, including LF Networking's OPNFV Verification Program (OVP) initiative.

Results

Since 2017, the use of ONAP has expanded to automating numerous key network services across all Bell business units. Moving forward, ONAP is playing a major role in 5G and multi-access edge computing (MEC) rollouts.

The key metric Bell uses to measure the success of ONAP is the number of recurring manual task hours saved per month. Each project that adopts ONAP for a specific service tracks this metric. In 2019 alone, Bell saved a significant amount of recurring manual work per month as a result of using ONAP. In 2020, the team will also measure the acceleration of new services onboarded to the platform. Currently, the onboarding process can range from a few weeks to six months. The NSO team's objective is to significantly reduce this timeframe by improving documentation and training materials, and making ONAP components easier to use.

Learnings and Summary

1. There is no need for an all-or-nothing approach to ONAP. It is better to deploy only the modules required.
2. In-house software development expertise is very important. As an open-source project, ONAP is at its best when its users also contribute to the code base and help to build the project's roadmap. The model is completely different from buying software and professional services from a vendor.
3. Kubernetes expertise is extremely important. The NSO team regularly encounters and fixes ONAP deployment, Helm chart and related problems.

The NSO team is also an active contributor to the ONAP roadmap in terms of providing requirements.

An early adopter of ONAP, the Bell team has learned how to work with the ONAP community and deploy ONAP in their network. Bell created an internal open source team, adopted DevOps, and moved to an on-site Reliability Engineering model to find continued success with ONAP over three years. The company is well-positioned full 5G and edge computing automation, while reducing operational expenses and improving the customer experience.