Network Orchestration –
A cornerstone in Virtual1's success

Virtual1 is a shining example of how the automation of day-to-day tasks within carrier networks, such as quotations, ordering, delivery and change management, can not only improve the overall operational expenditure, but increase both partner and end-customer satisfaction, leading to fast growing sales. In this interview, their Principle SDN Architect John Graham discusses how network orchestration has been one of the cornerstones of their success.

John, how has network orchestration contributed to the success of Virtual1?

At Virtual1, we use network orchestration to help us manage our partner network. This allows us to implement a wide range of customer services based on preconfigured service templates, and helps us centralise our resource management for IP pools, VLANs, AS numbers etc. Our service configuration is standardised and can be deployed rapidly to all affected elements of a network, without needing the input of an engineer to manually apply configuration via the CLI (command line interface).

Finally, any services we deploy are readily available for integration with other internal or customer facing platforms. This is done via a comprehensive API suite and makes our solutions much more seamless to operate. To achieve this we use PacketFront's multi-vendor network automation software, BECS.

How has this helped Virtual1 expand as a business?

A key feature of our offering is the control of ordering and provisioning processes for customers via our 1Portal. This control would have been greatly handicapped by a lack of a highly-flexible automation and a complementary API suite. By having this in place our customers are able to effectively reach into the Virtual1 network to carry out their own ordering and service provisioning, without the necessity of a constant chain of communication with various administration and technical departments. Bandwidth changes, the addition of new service modifications and additional VLANs for new service types can be performed by the customer at the click of a button via 1Portal which will then interact with our BECS platform and Salesforce to provide all necessary configuration and customer record updates.

Cost and timescales would also have been severely impacted by a less dynamic automation platform which is effectively ready 'out-of-the-box'. Where it might have originally taken half an hour to build a single internet service using a well-trained engineer working at maximum efficiency, with BECS we reap a sixfold increase in productivity as we can generally get this down to between 1-5 minutes, including the reservation of resources and updating of records.

As well as this, our previous platforms for managing resources have often been dated and thus, ran slowly. This adds to the time expended waiting for legacy systems to respond. All of these time overheads have now been eliminated.

How long did implementation take?

At the time of implementation our network consisted of MX PE routers and 50+ aggregation and edge switches, supporting 8000 business customers. It took 4 months to take the BECS platform through complete production and implementation, which included service templated design and coding, building of the server infrastructure and handing over management of the network to the BECS platform took approximately 8 months with a team of 3 engineers.

It is worth noting that even if the network was significantly larger this would not have added proportionately to the timescale for the project due to the highly scripted template format of service delivery.

How easily can you implement this when you already have an existing network?

BECS intelligently joins up the four key dimensions necessary for effective automation: Resource Management, Service Configuration, Topologies and Element Configuration. While the complete implementation is optimal, when incorporated as part of a new network with standardised service configurations, the flexibility of the platform has allowed us to integrate with an existing production network consisting of a wide variety of non-standard customer service configurations. Given sufficient development time, it is also possible to retroactively integrate older service configurations with service templates. Configuration sections can be excepted entirely from the control of the SDN platform or can be defined independently of any automated processes.

How has life as a network engineer changed within Virtual1?

Compounded with Salesforce integration, with BECS we now have one-touch provisioning, where a single button push can completely build out a customer service and prepare it for activation, while automatically updating the Salesforce record for that customer. This includes information such as PE unit numbers, reserved IP prefixes, hostnames, bandwidth etc. So, we spend less time configuring network equipment via the CLI. Other changes are largely committed from a GUI connecting to the BECS platform. This greatly reduces the likelihood of human error, allowing for greater efficiencies and enhanced productivity.

The engineers you have directly supporting BECS need to have both good networking skills and at least a basic understanding of scripting, and some of the aspects of working within a coding environment (git repos etc). Also, we still need to maintain and develop our technical skills such as troubleshooting, which requires the same diagnostic abilities as before. The system does not 'think' for the engineers. So consequently, we must expand our skillset to integrate an understanding of the SDN platform with our networking experience.

What were people's concerns when implementing/learning about the software? How did you overcome these?

The main concerns were how steep the learning curve would be, and how we can quickly adapt to changing network service design requirements to cope with new products being introduced in an ever changing marketplace.

The issue of learning how to operate BECS was resolved via a series of classroom learning for existing staff, along with a new starter training process. This was all developed internally. We have also extensively documented various customer service provisioning and modification tasks, along with troubleshooting guides for the technical teams. Because BECS is a low level platform controlled by an array of simple script files it is relatively straightforward to create new service definition templates, allowing us to remain both scalable and flexible enough to adapt to market changes.
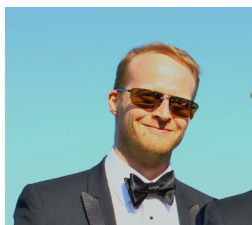
How was BECS able to help with Virtual1's expanding network?

BECS complements our rapidly expanding carrier network by allowing us to rapidly deploy all of our core elements. The software helps to support automated configuration and resource management within a single contiguous platform. Equipment can be shipped to site and connected via remote hands operators without any prior configuration required by our staff, and through the use of the zero touch automation functionality within BECS.

What's the main business benefit we get from the software?

The main benefit we get from the software is scalability, with an ever-changing network environment and improved efficiencies due to reduced likelihood of human error and a much faster process for provisioning new and modifying existing services.

BECS can seamlessly integrate with existing or new internal systems to easily support components such as capacity management, customer orders, monitoring platforms.

Learn more about the impact of automation at Virtual1 — visit our page.

Barnhusgatan 20
P.O. Box 575
SE-101 31 Stockholm, Sweden

Sales +46 8 633 1991
Switch +46 8 633 1990
sales@pfsw.com

www.pfsw.com