MOOR INSIGHTS & STRATEGY

APSTRA'S AOS: DISTRIBUTED NETWORK OS

A NETWORK ABSTRACTION LAYER COULD BE A BLUEPRINT FOR FUTURE DATACENTERS

EXECUTIVE SUMMARY

Networks today are struggling under the weight of 25 years of proprietary technology and manual processes. Customers either have been tied down to a single vendor in order to ensure interoperability or have been forced into integrating and maintaining multiple vendors' equipment on their own, with plenty of headaches and finger pointing when things do not run as anticipated.

With the current need to move quickly and flexibly, customers are beginning to learn that their network is holding them back. There is no single vendor providing a solution that enables customers to buy the exact product that suits all their needs while at the same time ensuring full interoperability with the rest of their environment. Apstra is taking a new approach to resolve this challenge with a strategy that creates the basis for distributed network automation. The Apstra Operating System (AOS) is Apstra's solution to these challenges: a distributed operating system for the network. AOS enables an enterprise to bridge the gap between configuration and operational state, closing the loop and providing a platform to deliver fully-automated network services. By pulling all of these elements together into a single solution, Apstra creates an abstraction layer that enables a single, continuous view into the network.

TODAY'S NEEDS

Customers have always had to walk a fine line between purchasing the products that exactly meet their business needs or purchasing the products that will fit in with their current environment or management structure. Because of the high cost of managing networks and the operational challenges that ensued, customers were more likely to buy new products based on what was already in the rack than choose alternatives based on what was truly the best product to meet their needs. This captive customer environment slowed innovation and left customers with less flexibility for deploying network services in support of their growing businesses.

Today's networks eventually end up as bottlenecks for customers. Opportunities can be missed simply because the company cannot move quickly enough, with network configuration typically being "the longest pole in the tent" for making changes. Network change requests pile up, and critical tasks are not handled because of the many manual processes and complexities that administrators face. Adding additional vendors intensifies these problems, as each brings its own unique management tools and processes.

Networks have never been designed around applications. Instead, because of network limitations, applications are typically modified to fit into existing norms and strategies. For administrators, there is an eternal struggle between the ideal (as defined in the configuration) and the reality (the actual state of devices). To most customers, a multi-vendor strategy adds risk as it multiplies all the issues (and operational costs) that admins were already facing with a single vendors' products.

The promise of open networking is a big lure for customers today, but making the transition is difficult; although there are standards, much work needs to be done to integrate all of these components. The lack of a common language across the network is real. There are standards in networking (IP, BGP, *etc*.) that are universal and well-defined, but the vendor-specific configuration methods, each with its own flavor and tweaks, are a major cause of integration problems.

Customers are especially keen to solve these problems with a focus on implementation by **business intent** (not specific hardware features) backed by a continuous stream of telemetry and validation information. The goal is to help lower the risk of downtime and quickly debug situations if they arise. However, the challenge is that while an open network (like Facebook has built) is appealing, building such an environment for the typical company is resource-intensive. Not having appropriate resources (or budgets) has left CIOs desiring this type of change but lacking infrastructure or resources to get there. Changing to a more flexible environment requires viewing networks differently.

APSTRA CHARTS A DIFFERENT COURSE

As many networking vendors struggle with stemming customer defections and securing their walled gardens, Apstra has entered the market with a different approach. Instead of competing with them, Apstra has created an environment that enables these vendors to continue their direction and yet still engage in a broader strategy for bringing all the networking components together in a unified fashion. Apstra was founded and built by network technology leaders and has a team of experts from companies such as Arista Networks, Cisco Systems, and Juniper Networks, as well as a close tie to the leadership of the Stanford Distributed Systems Group. Their expertise in both network engineering and distributed systems gives Apstra the knowledge to drive innovation.

In a staid and conservative market like networking, the idea of disruption is typically viewed with some skepticism. But the reason for slow change—the complexity and personality of the products—is exactly what Apstra aims to address by creating an abstraction layer that nullifies many of these challenges. To be successful, a networking vendor needs to have a pervasive technology that also heavily leverages standards. Apstra recognizes that the network is an inherently distributed system and that the industry needs a product that automates network services as a distributed system. With AOS, Apstra is addressing the market through public APIs and access methods. This strategy creates a layer of abstraction that allows Apstra to bypass so much of the noise in the market, creating a laser focus on customers' real needs.

Early customers are already finding success with Apstra. For instance, a SaaS / cloud customer that is engaged with Apstra has a handful of network engineers managing more than a dozen datacenters with a collection of Cisco hardware and Cumulus software. They needed a vendor-agnostic solution that allowed them to be more agile. A large financial customer also came to Apstra after struggling with a multi-vendor network (Cisco / Arista Networks). They needed to simplify operations through automation, reducing their costs and improving their network availability.

Apstra talks about creating the "operating system for the network". But what does this really mean? Clearly, network devices all have operating systems (like Cisco NX-OS and Juniper Junos OS), but this analogy goes much further. Apstra's goal is to create an environment that brings the attributes of an operating system to the **entire network** as a single entity. Key elements of operating systems include:

- Drivers that allow for the interface of multiple devices and applications
- Application services layers that enable the OS to service applications horizontally
- Resource tracking and connection arbitration
- Tracking all revision levels of all components and the ability to automate, keeping these all up-to-date and running smoothly

The PC world has moved to a shared model where a GUI OS manages and maintains interfaces and services for all applications. The network needs to do the same.

Switch vendors are attempting to create an open, device-level OS, while others are attempting to boil the ocean with a universal device-level API. Beyond the war of CLIs and more recently APIs, Apstra is tackling automation at a higher level by creating an operating system of the network. This extensible design could go far beyond the network and become, as Google has described, the operating system of the datacenter.

The key to success here is creating an environment that is vendor-agnostic, enabling all vendors to join in while simultaneously enabling customers to leverage their existing investments (and avoid the forklift upgrade). Apstra is working with all of the leaders in networking to achieve this harmony across multiple platforms.

AOS Delivers Distributed Network Automation

The vision for AOS is to deliver **intent-driven** datacenter automation through a distributed operating system that is vendor-agnostic, enabling a business to abstract all of the network functions and manage them as a single entity. AOS has an intent-driven applications focus that zeroes in on "*what did you mean*" instead of "*what did you say*".

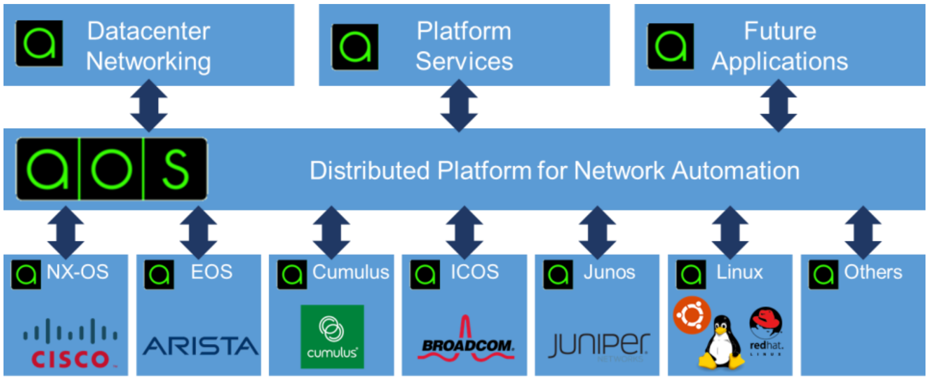
In traditional networking, well-regarded standards like BGP, OSPF, and LLDP are combined with the proprietary glue that vendors use to differentiate their products. As much as possible, AOS leverages those industry standards while creating an abstraction layer that enables everything to be managed together regardless of the vendor, which is a key demand for customers.

Because a vendor's primary responsibility is supporting its own products, network automation equates to "automation of our products" and not "automation of customer needs". The device-centric view is very much a bottom-up approach, while customers view things top-down. AOS creates an abstraction layer that is not tied to individual vendors' products but instead to the customers' needs.

With a lifecycle approach, AOS can begin with the customer intent as it builds out its three key artifacts: the **configuration** (desired state), the **operational state** (reality), and the **alerts** (the discrepancy between the desired state and reality). With AOS, intent provides a context for automating alert generation. In other words, alerts are built-in and remove the need for traditional rules-based integration that is both fragile and costly. Through real-time telemetry, analytics, and visualization that all tie back to intent, administrators can be notified immediately of changes and other problems occurring in the network and see the impact of these occurrences in a before-and-after view. This feedback allows them to focus on keeping the network running and ready for business. AOS's automation workflows build value over time as the system learns more about a network and can fine-tune its parameters to match the real-life environment.

A truly extensible turnkey solution needs to have extensive customization capabilities to ensure high productivity, enabling the customer to focus their DevOps resources on areas with the highest value. The three artifacts (configuration, state, and alerts) can all be customized to fit into customers' existing environments and processes. This customization enables an administrator to become productive immediately versus having to learn an entirely new structure or language to get up to speed.

FIGURE 1: THE APSTRA AOS ECOSYSTEM



DISTRIBUTED OPERATING SYSTEM

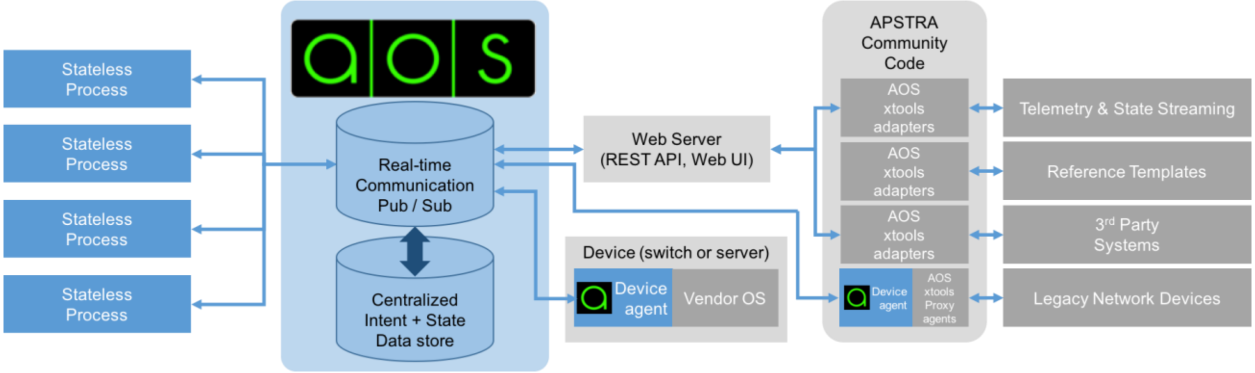
A truly distributed operating system needs a distributed data store (as well as automation of the common platform services) to provide management of the resources, devices, and workflow. AOS uses these components to build a distributed OS that optimizes performance while simultaneously improving availability. Apstra's distributed nature removes the typical network configuration model points-of-failure.

As a distributed operating system, AOS does much of the heavy lifting when addressing the most difficult problems. Once these are taken care of, all of the AOS applications can leverage this work to run more easily. Problems that AOS addresses include:

- Managing large distributed states
- Maintaining a state-driven logical publication / subscription (pub / sub) channel for communication between components
- Managing networking resources
- Maintaining all of the device abstractions
- Providing extensibility (being non-opinionated about how services are designed)

Also, AOS spans both physical and virtual devices. For example, AOS can enable a datacenter networking application that automates IP fabric connectivity across both the bare metal platform and the virtual machines running on top of those systems. AOS is built around standards, but if device-specific protocols are required for specific device implementation, then these can be programmed into AOS to help integrate these very specific components into the network. This programmability allows AOS to be the abstraction layer, arbitrating between these components and the rest of the network.

FIGURE 2: APSTRA AOS ARCHITECTURE



CONTINUOUS SERVICE AUTOMATION

AOS automates device configuration and provisioning based on actual business intent, not some arcane command line input or device-level manipulation. This continuous service automation enables ongoing management and maintenance based on intent, enabling AOS to adjust the network dynamically. This continuous verification and feedback cycle helps the distributed operating system narrow down actions, aligning them with intent and fine-tuning them with each cycle.

DISTRIBUTED DATA STORE

The distributed data store, by its nature, provides for scale and redundancy while still providing a single view of the whole system. This data store is the repository of all of the operational states in the system as well as the business intent. The combination is the hallmark of what is required for something to be a true operating system. The distributed data store describes intent, which allows for the building of applications that can automate various network services or use cases. Additionally, the system is able to abstract out all the devices through the use of device drivers, which are optimized to interface with each of the individual components. These device drivers help manage configuration and operational state through separate interfaces, so abstractions remain manageable, avoiding the complexity explosion that plagues traditional approaches.

As the deployment grows over time, the data store becomes larger, more integral to operations, and of course, more valuable. This is the opposite of a device-level OS that becomes less valuable as deployments grow because it is tied to a single device and managed individually instead of being leveraged across the whole enterprise.

While the distributed data store is responsible for managing and maintaining all of the configuration and state information, it is far from being just a passive repository. By managing and maintaining this configuration and state information, the data store becomes the focal point of AOS. Through real-time communication, facilitated in the pub / sub process, the system keeps everything up-to-date, managing the elements and changes to keep the business online. Because of the open nature of AOS, 3rd-party applications and tools can be created easily to access the data store through the REST API. This functionality enables integration into 3rd-party systems as well as supporting legacy networking devices.

A Service-Oriented View

To develop a truly services-oriented view, continual data feeds from every physical and virtual managed resource across the infrastructure are fed into the system. The continuous input of telemetry data helps the system zero-in on the status of the network and how the different service components are performing. Information from both the physical and virtual layers are unified in the AOS Distributed Data Store, so they can be correlated natively without relying on external management tools. This correlation allows administrators to discern the total state of the network and whether any corrective actions need to be taken. Administrators can observe correlation between various pieces of network configuration and network state without having to correlate these components themselves. For example, if a transceiver fails, then AOS will not only notify the administrator of this failure, but it will also outline the resulting consequences of this failure (*e.g.*, ensuing link failures, protocol layer failures, connectivity problems affecting specific hosts or routers). In another example, AOS gives administrators correlation information between the physical and virtual layers, enabling them to correlate incidents in one layer with related occurrences in the other.

Device Management, Resource Management, & Other Platform Services

The AOS Distributed Operating System provides platform services that are common across all applications. Maintaining a distributed data store entails both registering and then inventorying the devices on the network, a function that AOS automates to ensure that all the devices are identified and stored properly. The AOS platform implements resource pools which, through documented user-facing REST APIs, allows for integration with 3rd- party applications and tools. The goal is to allow organizations to integrate their existing system (such as IPAM and other resource management tools) seamlessly with AOS. Customers will have the ability to define their own standards and practices and, more importantly, also have the ability to apply industry best practices. The capability to customize allocation patterns or algorithms based on specific business needs increases the value and flexibility of AOS for the customer.

THE CUSTOMER PROCESS

To begin the process, the turn-key AOS Datacenter Network application is deployed, and all the devices are discovered and registered. The distributed data store is populated with all the device information, creating a starting point. Network services can now be built based on business intents. Unique service instances as well as more generic, horizontal instances, can be created. These services are then deployed into the network where they are monitored and managed. As conditions in the network change, AOS can notify administrators or make the appropriate changes to the deployed services, logging the updates back to the distributed data store for future analysis. An easy way for customers to get started is to download the AOS virtual instance from the Apstra website and run it in a sandbox virtual environment, managing virtual device instances without the need for having the actual hardware connected.

Competitively, no other companies are doing the same thing as holistically as Apstra. AOS has created a new vendor-agnostic product category for network automation. Other automation tools are available, but these are typically provided by a vendor and focused on only that specific vendors' own products, not the wider market. Although these often have a way to pull in other products, they generally do not offer the same level of robust capability across others' products versus their own. Additionally, the products available today are typically rules-based instead of being intent-based. Rules-based capability is the lower-hanging fruit of network automation, because it relies on a single vector: if x = y, do this. AOS is different, because it focuses on the intent and then can make changes that reflect actual conditions and impacts, aligning services back to the original intent. Network switch vendors are beginning to create automation layers but mainly at the switch level, not extending across the rest of the network functions and services, and not truly intent-based.

The architecture that Apstra has built includes the capability to move beyond just managing the network. There is no reason to believe that a distributed operating system for networking devices and services could not be used for other devices and services within the datacenter. Thus, what AOS is today could theoretically grow into something that more closely resembles an operating system for the whole datacenter, not just an operating system for the network. This concept is important as we look at how integrated networking has become in the applications and architectures deployed today, especially in the realms of the Internet of Things (IoT) and big data / analytics. These are areas where networking—because of the heavy application data transfers—plays a key role in the overall performance of the applications.

CALL TO ACTION

Customers today who are struggling with maintaining their current network and who believe there should be a better way to automate and manage it should consider Apstra's AOS as an alternative to their current tools. With the need to make networks move and adapt at the speed of business, customers need to look beyond the old rules-based alternatives and towards an intent-based environment that will help the network move and adapt quickly to business needs, instead of being the bottleneck.

Apstra is redefining the networking landscape by introducing the first truly distributed network automation through AOS. This distributed network automation has the potential to extend beyond just network devices and services, making it possibly more pervasive within the datacenter. We recommend that customers investigate Apstra as an alternative to not only the competitive automation products on the market but also as a new functional direction for automating and managing the network services to better align them with the growth needs of their company.

IMPORTANT INFORMATION ABOUT THIS PAPER

AUTHOR

John Fruehe, Senior Analyst at Moor Insights & Strategy

PUBLISHER

Patrick Moorhead, President & Principal Analyst at Moor Insights & Strategy

EDITOR / DESIGN

Scott McCutcheon, Director of Research at Moor Insights & Strategy

INQUIRIES

Contact us if you would like to discuss this report, and Moor Insights & Strategy will promptly respond.

CITATIONS

This note or paper can be cited by accredited press and analysts, but must be cited in-context, displaying author's name, author's title, and "Moor Insights & Strategy". Non-press and non-analysts must receive prior written permission by Moor Insights & Strategy for any citations.

LICENSING

This document, including any supporting materials, is owned by Moor Insights & Strategy. This publication may not be reproduced, distributed, or shared in any form without Moor Insights & Strategy's prior written permission.

DISCLOSURES

This paper was commissioned by Apstra. Moor Insights & Strategy provides research, analysis, advising, and consulting to many high-tech companies mentioned in this paper. No employees at the firm hold any equity positions with any companies cited in this document.

DISCLAIMER

The information presented in this document is for informational purposes only and may contain technical inaccuracies, omissions, and typographical errors. Moor Insights & Strategy disclaims all warranties as to the accuracy, completeness, or adequacy of such information and shall have no liability for errors, omissions, or inadequacies in such information. This document consists of the opinions of Moor Insights & Strategy and should not be construed as statements of fact. The opinions expressed herein are subject to change without notice.

Moor Insights & Strategy provides forecasts and forward-looking statements as directional indicators and not as precise predictions of future events. While our forecasts and forward-looking statements represent our current judgment on what the future holds, they are subject to risks and uncertainties that could cause actual results to differ materially. You are cautioned not to place undue reliance on these forecasts and forward-looking statements, which reflect our opinions only as of the date of publication for this document. Please keep in mind that we are not obligating ourselves to revise or publicly release the results of any revision to these forecasts and forward-looking statements in light of new information or future events.