TECHNOLOGY BRIEF

Intent-Based Analytics™

Detect and Prevent Network Outages and Gray Failures

Dynamically and quickly extract insights from raw telemetry

- **Declare your network intent (e.g. connectivity, oversubscription, isolation, redundancy)**
- **IBA validates your intent continuously, alerts you with intent deviations instantly**
- **IBA extracts contextual and actionable insights from raw telemetry in minutes**
- **Eliminate complex and brittle rule programming**

Intent-Based Analytics™ (IBA) is automated big data analytics designed to cut data center network outages and gray failures by at least 50%.[1] It liberates network operations from consuming raw telemetry, writing and maintaining complex and brittle rules to keep up with ongoing network changes, and staring at network visualizations 24x7 in order to detect unusual patterns.

IBA is embedded in AOS®, an intent-based distributed operating system. IBA solves both the raw data collection and insight extraction challenge. Using a simple, dynamic, declarative interface, you can specify exactly how you expect the network to operate — beyond mere connectivity — and including traffic patterns, performance, and tolerance for gray failures.

IBA then continuously validates your intent, simply generating anomalies when it detects a deviation. With IBA, you can quickly detect and prevent a wide range of service level violations - including security breaches, performance degradations, and traffic imbalances (Figure 1). Most importantly, IBA works continuously in the presence of any change in network policies, configs and the live state, by recalculating your Service Level Indicators (SLIs) dynamically.

IBA: Turnkey, Multi-vendor, Closed-loop, Automated Big Data Analytics

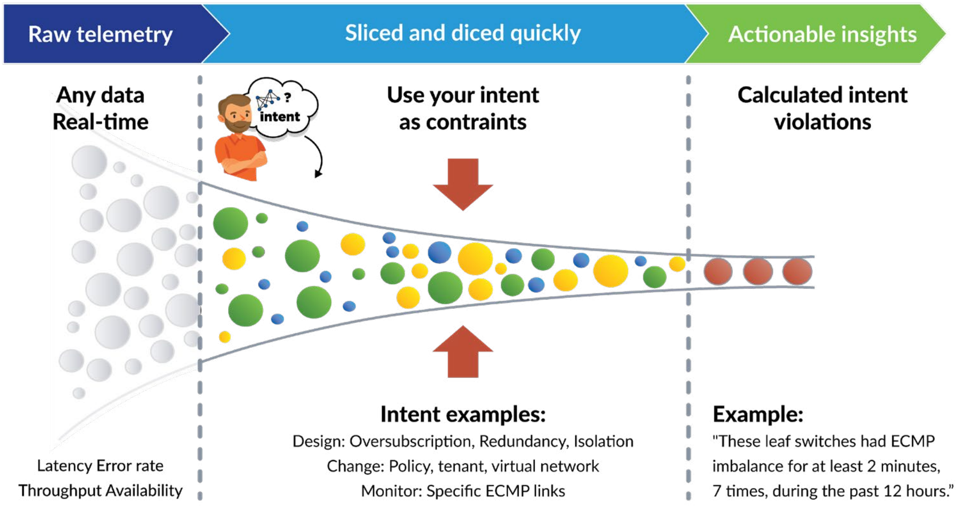

Raw telemetry
Sliced and diced quickly
Actionable insights
Any data
Real-time
intent
?
Use your intent
as contraints
Calculated intent
violations
Intent examples:
Design: Oversubscription, Redundancy, Isolation
Change: Policy, tenant, virtual network
Monitor: Specific ECMP links
Example:
"These leaf switches had ECMP
imbalance for at least 2 minutes,
7 times, during the past 12 hours."
Latency Error rate
Throughput Availability

Figure 1: Telemetry-to- insights in minutes

[1] *Gartner "Innovation Insight: Intent-Based Networking Systems", Published: 7 February 2017 ID: G00323513, Analyst(s): Andrew Lerner, Joe Skorupa, Sanjit Ganguli*

Without IBA: Your network analytics are subject to change whenever the network changes

With the traditional network management approach, you need to specify what raw telemetry to acquire, how to acquire them, and how to extract actionable insights from them. You also have to update your data acquisition and analytics pipelines whenever there is a change in the network. You have to constantly worry about:

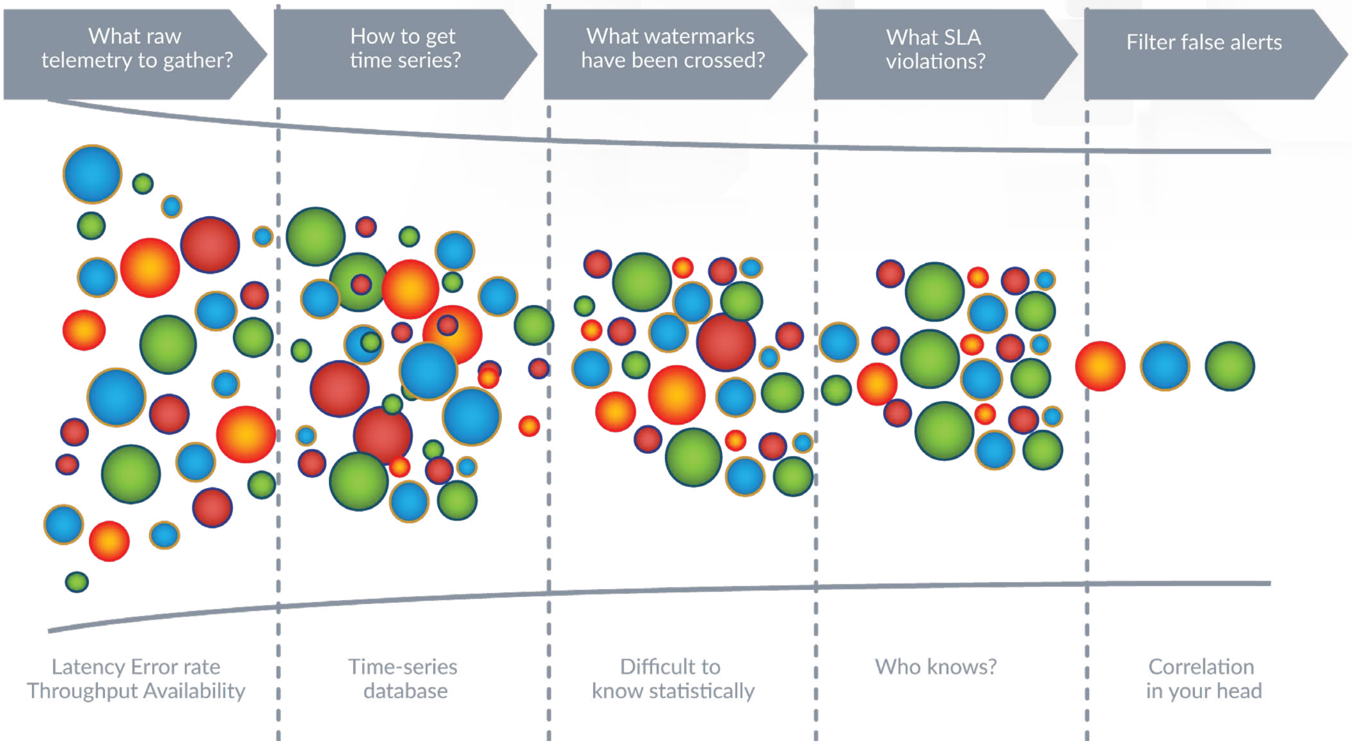

What raw telemetry to gather?
How to get time series?
What watermarks have been crossed?
What SLA violations?
Filter false alerts
Latency Error rate Throughput Availability
Time-series database
Difficult to know statistically
Who knows?
Correlation in your head

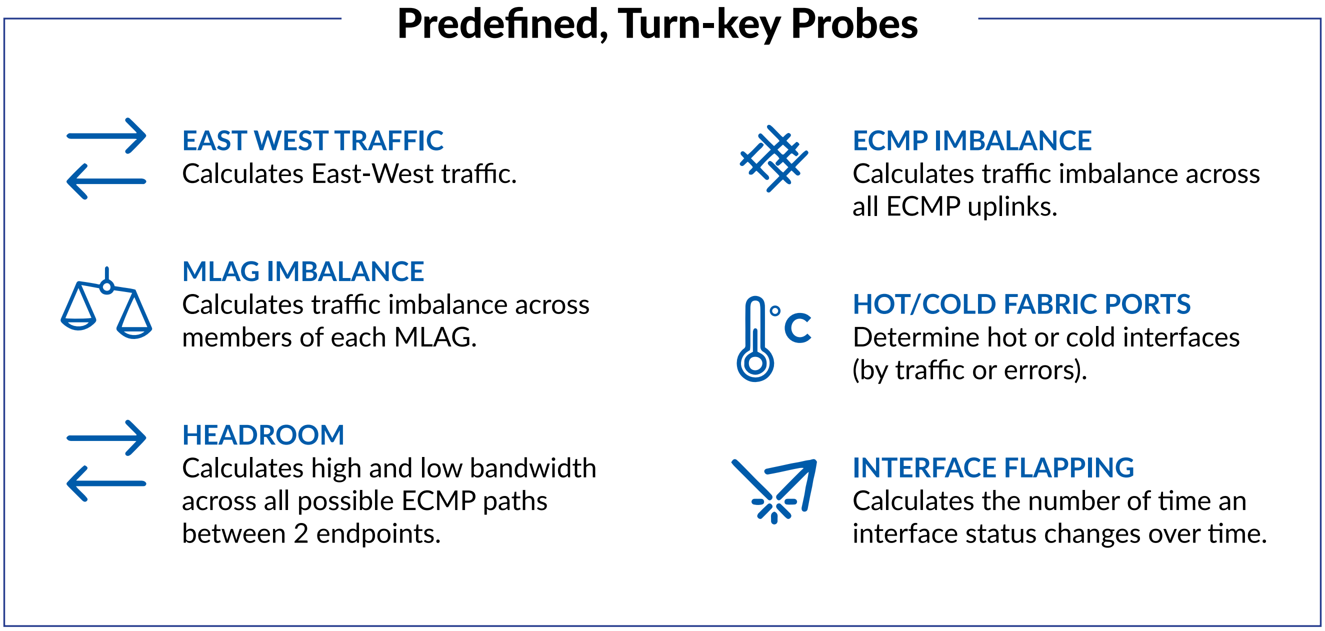
Figure 2: Traditional big data dump, slow convergence to insights

- What fabric links to monitor;
- What raw telemetry you need to measure latency, error rate, throughput and availability for each interface;
- Vendor-specific procedures and methods to gather raw data, what vendors SNMP MIBs you need, which are often late to market, still just sampling not real-time;
- How to integrate real-time telemetry streams, to find those transient and intermittent problems, using expensive tools to collect, parse, format, filter, stream, visualize and store raw telemetry;
- How to write complex rules to aggregate and calculate data across many devices and links into Service Level Indicators (SLIs), so you and executives can understand.

Essentially, you have to correlate the actual network state (raw telemetry) and the desired state of the network (e.g. your intent in oversubscription, redundancy and isolation). You integrate and correlate design diagrams, config files and show command dump in your head.

IBA: automated, closed-loop big data analytics without coding

IBA automates both the data collection and insight extraction process. It provides 6 intelligent probes out-of-the-box, allowing you to quickly set up proactive closed-loop monitoring and actionable alerts for gray failures that might lead to widespread outages. Use the following predefined turnkey probes out-of-the-box, and immediately cut your Mean-Time-to-Insight (MTTI) on those difficult-to-find fabric related problems.



Predefined, Turn-key Probes

EAST WEST TRAFFIC
Calculates East-West traffic.

MLAG IMBALANCE
Calculates traffic imbalance across members of each MLAG.

HEADROOM
Calculates high and low bandwidth across all possible ECMP paths between 2 endpoints.

ECMP IMBALANCE
Calculates traffic imbalance across all ECMP uplinks.

HOT/COLD FABRIC PORTS
Determine hot or cold interfaces (by traffic or errors).

INTERFACE FLAPPING
Calculates the number of time an interface status changes over time.

Figure 3: Default IBA probes

These probes work continuously in the presence of change, by recalculating your Service Level Indicators (SLIs) dynamically. And you can customize and create your own probes.

These changes can be frequent, ephemeral and hard to track. At the workload level, workloads are moved around due to failures and optimization.

At the network level, policies, tenants and virtual networks could change frequently. For example:

- Your intentional change in Service Level Objectives (SLOs, such as oversubscription 3:1, tolerance of MLAG and ECMP imbalance, isolation via L3 and L2), capacity change by adding, removing or replacing devices.
- Accidental change due to configuration drifts and device failures and software bugs.

In general, these changes can be anything throughout the lifecycle of designing, building and operating a spine-leaf network (table 1).

Your change of intent	Examples of change	Examples of dynamically updated IBA probes
Design intent	Change in oversubscription goals, redundancy, and isolation	▪ Detect link traffic imbalance between leaves and spines
Build intent	Change in vendor devices	▪ Detect when links are reaching saturation
Operational intent	Change in policies, virtual networks, and tenants	▪ Compare east-west and north-south traffic distributions
		▪ Detect MLAG pair traffic imbalance
		▪ Detect interface error/discard counters
		▪ Detect interface flapping
		▪ Compute available bandwidth between servers or switches.

Table 1: IBA works dynamically in the presence of any change in the network

IBA is vendor-agnostic. It works when you have different vendor devices and network OSes interoperating together in the network.

IBA: Embedded in AOS

IBA is part of AOS, an intent-based distributed operating system that helps network engineers and operators to design, build and operate data center spine-leaf networks at speed and scale.

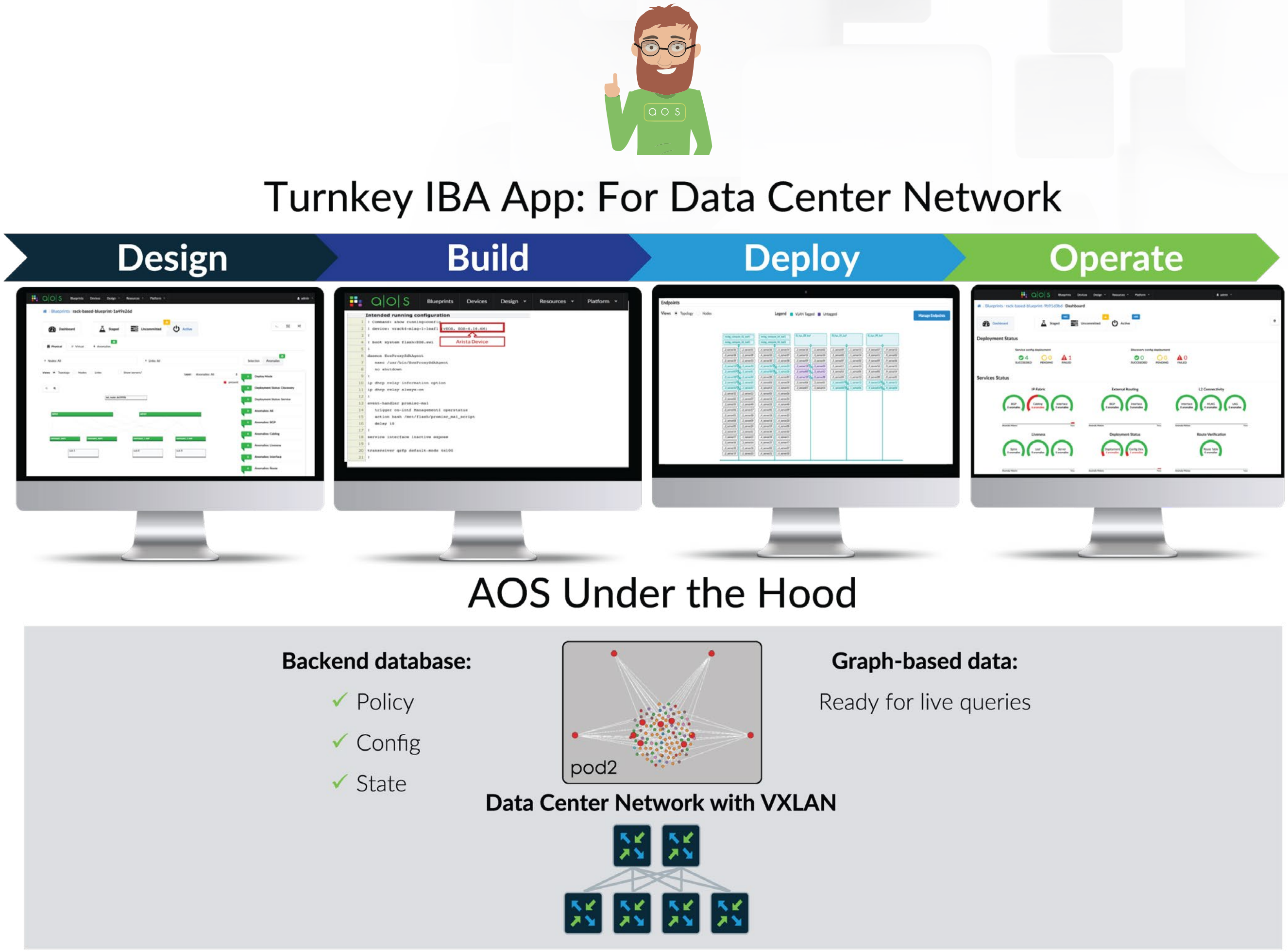
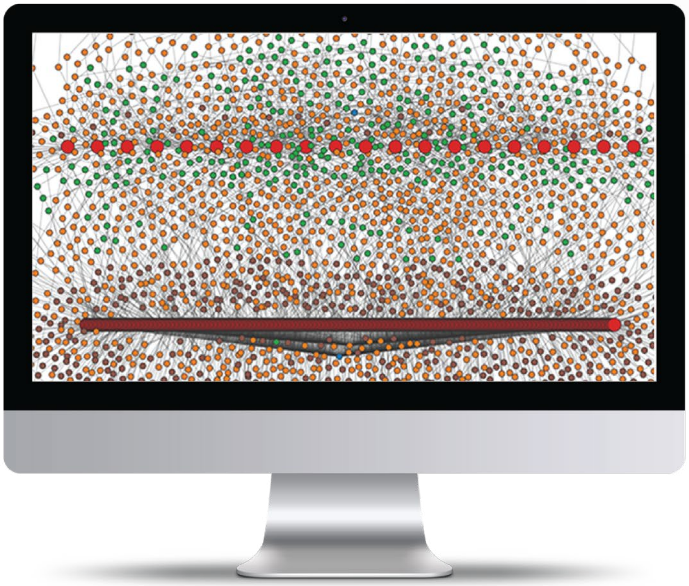


Figure 4: IBA and AOS

Because AOS integrates all your management activities and data across the design-build-deploy-operation stage, your policy data (connectivity and segmentation including VLAN and VXLAN), config data and operating state data are stored and correlated in one place, you can perform live queries to extract the insights you need, vs. having to gather and interpret raw data yourself (Figure 4).

“The following AOS screenshot shows a graph, a visual representation of the logical relationships within a large spine-leaf network, including relationships between servers, leaves, spines, links, interfaces, BGP peerings and more. The diagram is rendered by AOS based on its graph database, storing your SLOs, configs and best-practice SLIs and watermarks. The question on the right is a live query example.



A live query example:

Which ECMP links have imbalance for at least X minutes, Y times, during the past Z hours?

Figure 5: IBA Live Query Example

About Apstra®

Apstra pioneered Intent-Based Networking and Intent-Based Analytics™ to eliminate the complexities and inefficiencies that plague data center network operations today. Apstra's core mission is to deliver on the vision of a Self-Operating Network™ that delivers log scale improvements in CapEx, OpEx and capacity. Apstra was founded by leading experts in networking and abstraction (Arista, Juniper), distributed systems and automation (Google, VMware, Stanford). The company is privately funded and based in Menlo Park, California.

For more information, visit www.apstra.com, contact sales@apstra.com, or follow @ApstraInc