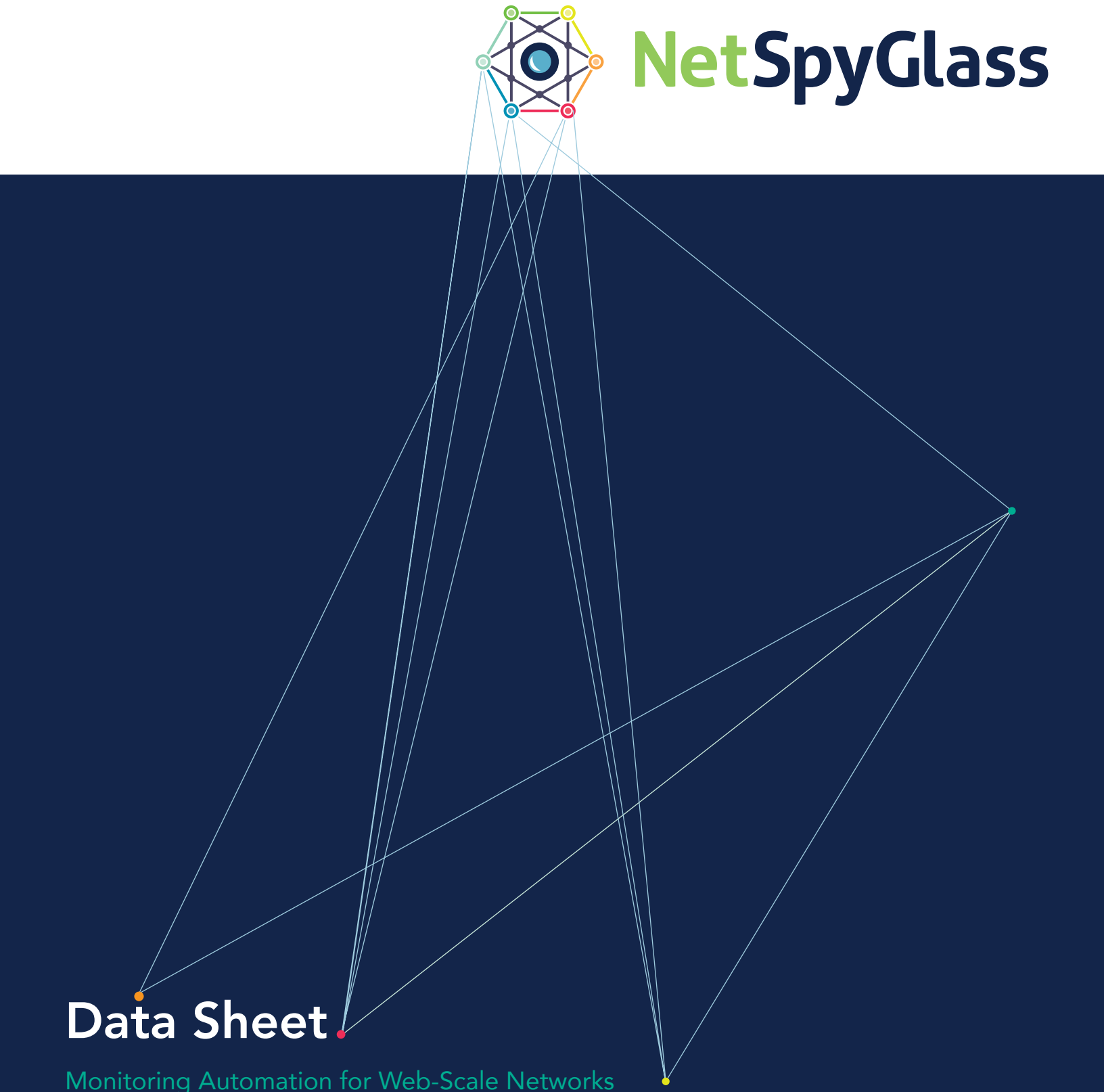
NetSpyGlass



Data Sheet

Monitoring Automation for Web-Scale Networks

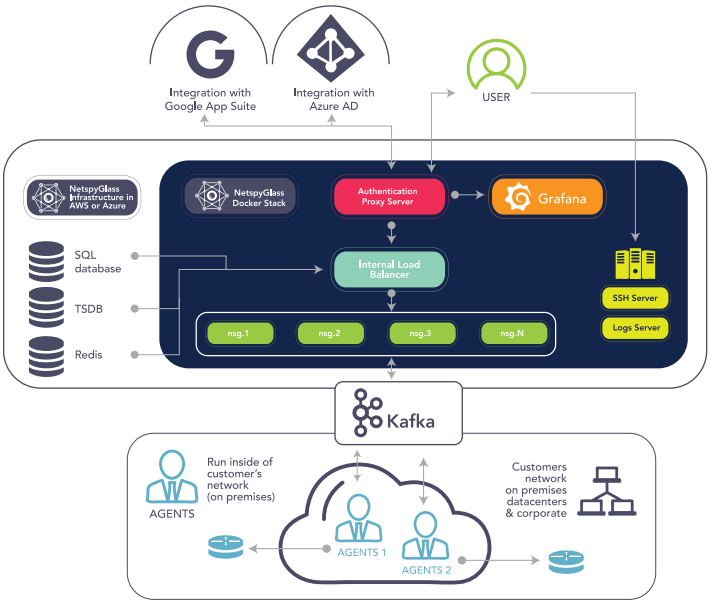
CLOUD-BASED MONITORING AUTOMATION FOR WEB-SCALE NETWORKS

NetSpyGlass (NSG) is cloud-based, network monitoring automation for network operators seeking an alternative to an unmanageable patchwork of tooling. As a SaaS (software-as-a-service) solution, NSG requires no up-front investment for network operators to gain real-time network monitoring, mapping, visualization, alerts, analytics and reporting. NSG is unique as a network monitoring solution in terms of its scalability, programmability and usability. It is designed to monitor many thousands of devices while collecting millions of metrics per minute. It hosts an embedded Python interpreter enabling automation of complex monitoring workflows. And, its browser-based UI was crafted to make NSG easy to use with minimal maintenance and maximum productivity.

Key Features

- Automation: Leverage embedded Python interpreter to automate building network maps, detecting and responding to changes or triggering actions and alerts.
- Scalability: Accommodate thousands of devices, hundreds of thousands of interfaces and millions of metrics through scalable cluster configuration.
- Auto-Discovery: SNMP v1,2,3 device polling to automatically discover vendor, model, protocols, component inventory, configuration, interfaces and more.
- Root Cause Alerting: Alerts operate on NetSpyGlass collected monitoring data which is evaluated against user-defined conditions to trigger notifications.
- Real-Time: Gain granular visibility into network state; collect, process, store monitoring data and see parametric values in context as map overlays.
- Historical Maps: Visually display historical monitoring data in the context of network state and topology as devices/connections change over time.

NetSpyGlass Service Architecture



Key Benefits

- Monitor massive (i.e. web-scale) networks with a highly consolidated toolset enabling efficient/effective workflows
- Eliminate errors caused by manual configuration with fully automated discovery of heterogeneous, multi-vendor networks
- Minimize risks of downtime due to poor visibility and/or inadequate tooling
- Eliminate "alert fatigue" and associated operational risks using NSG's innovative (root cause) alerts & notifications
- Optimize network performance using insights gained through programmatic flexibility and rich analytics
- Reduce costs with minimum on premises footprint with NSG's entirely cloud-based infrastructure (i.e. no CAPEX)
- Satisfy 80% of (Network Operations) requirements (out-of-the-box) with no special operator configuration required
- Reduced operator intervention and ongoing maintenance efforts for network monitoring; just add devices and NSG knows what to do
- Monetize network monitoring with NSG's open API enabling queries on live data for analytical processing in external systems
- Future-proofed network monitoring designed to be ready for SDN (Software-Defined Networking) and IBN (Intent-Based Networking)

NetSpyGlass

NetSpyGlass "Agents"

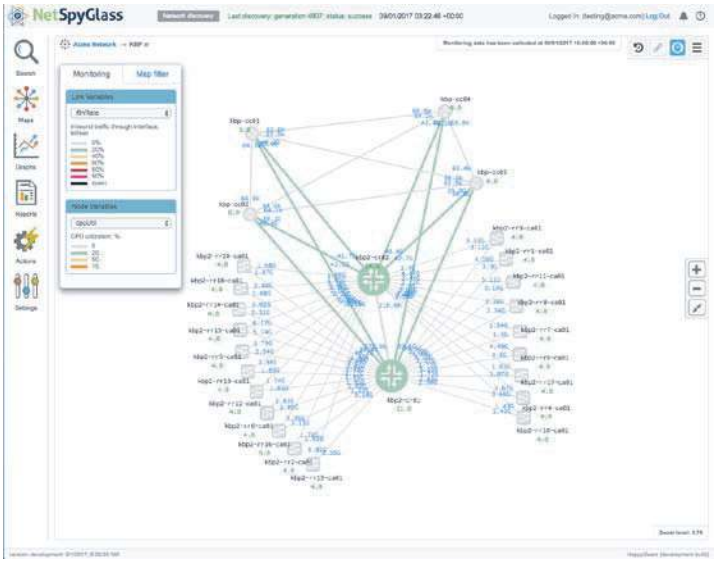
- NetSpyGlass requires installation of Linux-based “Agents” on the user's network.
- NSG Agents are very light-weight extensions of the core NetSpyGlass monitoring engine.
- Agents are responsible for maintaining device, interface and hardware component visibility and for communicating data back to the NSG core monitoring engine.
- Agents can be hosted on a virtual machine (VM) or on a shared machine and do not require dedicated hardware.
- A typical installation would deploy several NSG Agents to ensure good performance and to provide additional redundancy. The total number of Agents that should be deployed depends on the network size and architecture, but 500 - 1000 devices per agent is a common practice.
- If the network is geographically distributed, deploying a pair of Agents in each geographic region is recommended even if the total number of devices is not very high.
- As a general rule, Agents should be deployed in close proximity to the devices they will monitor to minimize potential effects of network latency.

Automated Network Discovery & Topology Mapping

NetSpyGlass discovers Layer2 network topology using information collected from devices, ranked in the following order of dependability:

- LLDP
- LAG (802.3ad)
- CDP
- Switch MAC forwarding tables contents
- Spanning Tree state
- ARP tables
- IP-IP and IPSEC tunnels

Network topology is presented to the user in the form of interactive network maps. Users can build custom map views to show a subset of the discovered devices using various matching criteria. Devices can be selected (for display) using tags automatically applied during discovery or applied by the user. Devices can also be added to maps using any of the included (automatic algorithms) that enable users to add devices by just selecting check boxes, e.g. (add connecting devices) and/or (add adjacent devices).



Types of Data Collected

NetSpyGlass automatically discovers devices and very quickly retrieves a vast assortment of metrics and monitoring data "out of the box", i.e. with no special configuration. This includes but is not limited to the following:

- Basic device data such as uptime, its vendor and model, software revision it runs and so on
- Interface utilization, errors, discards, packet rates, speed, up/down status, duplex status
- Link Aggregation Groups: configuration (aggregation ports and aggregation interfaces), LACP state of aggregation ports
- QoS queues: RED and tail drops, queue utilization
- CPU utilization,
- Memory utilization
- Temperature
- Operational state of hardware components, e.g. fans, power supplies
- Optical transceivers: optical power on transmit and receive
- BGP sessions: peer AS, address, name; network interface responsible for the session; statistics such as BGP session state, number of updates per sec, numbers of accepted and rejected prefixes per second
- Chassis alarms
- Firewall counters, firewall statistics (such as session count) on Cisco ASA and Juniper SRX firewalls
- Load balancer stats, such as server pool size, vserver status etc (mostly for F5 at this time)
- ICMP: ping monitoring of devices, collects rtt and packet loss
- Routing table size (ipv4 and ipv6)
- Power Distribution Unit (PDU): bank and phase load

Embedded Python Interpreter

NetSpyGlass's automation capabilities result from an embedded Python interpreter that enable the programmable features of the platform.

Right out of the box, NetSpyGlass can perform discovery, mapping and most core functions with little or no programmatic intervention due to the many libraries, modules and scripts included in the product's standard configuration.

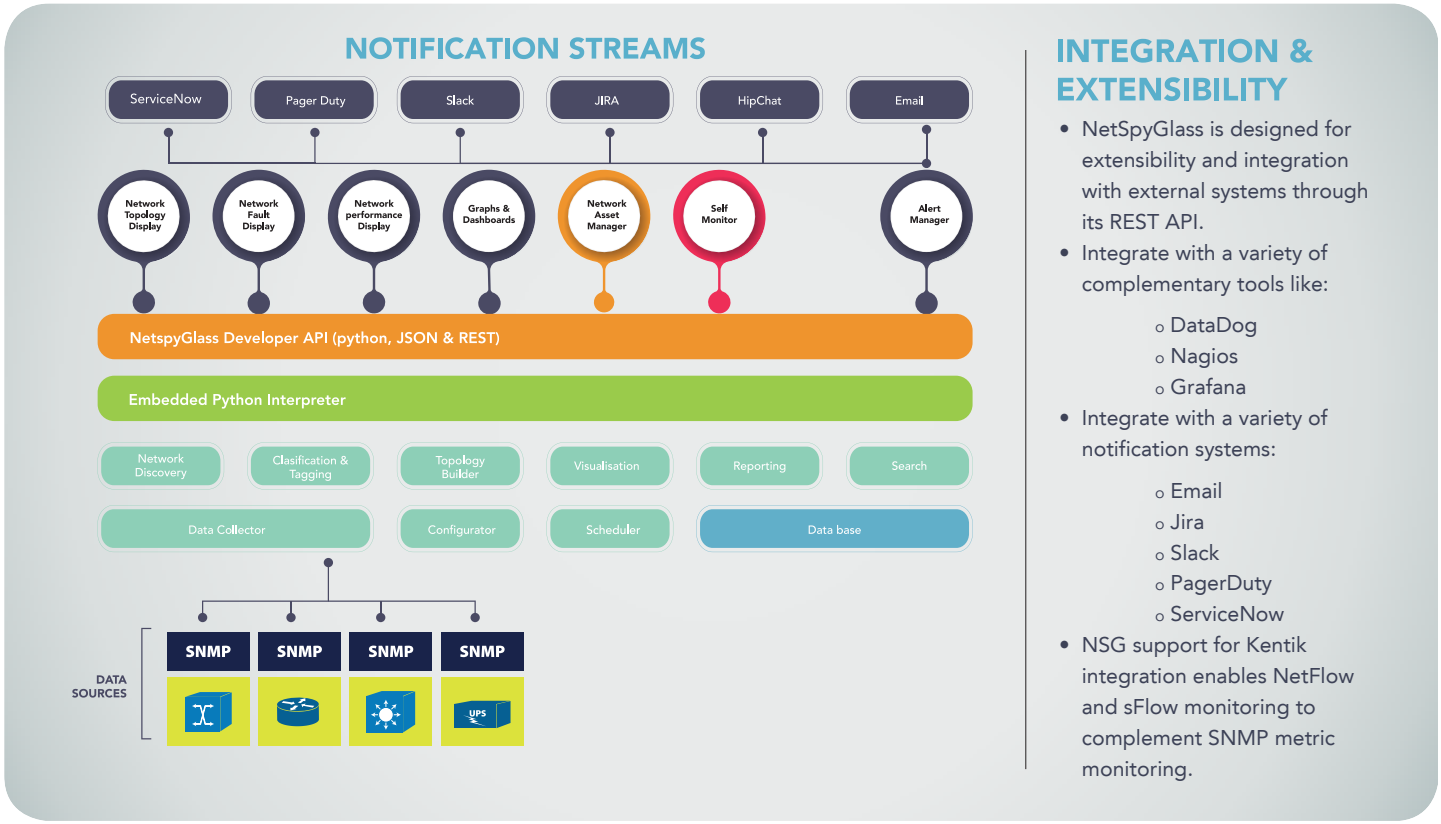
The embedded Python interpreter is accompanied by a collection of NetSpyGlass proprietary Python modules that implement a broad set of useful functions such as performing calculations with monitoring data (in Python) and generating new metrics.

NetSpyGlass Query Language (NsgQL)

NetSpyGlass incorporates a proprietary query language loosely based on SQL syntax (i.e. NsgQL) that can be used to select monitoring variables, devices and components. NsgQL can be used to build queries that access monitoring data, devices and components in NetSpyGlass. NsgQL queries also provide a very flexible mechanism for implementing complex logic to declaratively match by device, component and/or combinations of any tags to select the intended input data (see below examples).

Examples:

- SELECT device FROM devices WHERE Vendor=Cisco AND SoftwareRev="15.2(4)M2"
- SELECT device,interface FROM ifInRate WHERE BGP4Peer=AS1299



INTEGRATION & EXTENSIBILITY

- NetSpyGlass is designed for extensibility and integration with external systems through its REST API.
- Integrate with a variety of complementary tools like:
  - DataDog
  - Nagios
  - Grafana
- Integrate with a variety of notification systems:
  - Email
  - Jira
  - Slack
  - PagerDuty
  - ServiceNow
- NSG support for Kentik integration enables NetFlow and sFlow monitoring to complement SNMP metric monitoring.

Alerts & Notifications

NetSpyGlass alerts are implemented as small Python scripts. These scripts use the NsgQL language (see above) to select input variables. This scripting capability enables users to move far beyond static thresholds when designing an alerting system and to incorporate complex interdependencies between devices and interfaces and between physical and logical connections.

For example, users can build alerts to process time series data and use any of a number of functions to compare calculated values against thresholds. And, since alerts are implemented as Python scripts, users have access to all the expressive capabilities of Python as a modern programming language, including statistical analysis functions for anomaly detection. Some of the functions available to users to analyze monitoring data as part of an alert script include:

- SMA (sliding moving average)
- EMA (exponential moving average)
- Compare to the standard deviation to find outliers
- Function dbscan() to find outliers
- Function to find (steps) in the time series (i.e. big change in the value)

A key benefit of this capability is "root cause" alerting (as opposed to symptomatic alerting) that dramatically reduces or eliminates false positive and redundant alerts along with operator "alert fatigue".

Analytics

NetSpyGlass gives users powerful analytical capabilities by leveraging categorized (i.e. tagged) data to create custom metrics, aggregates or practically any kind of computational artifact.

- Use Python libraries and scripts for programmatic access to all data within the system for deep analytical insight
- Create custom metrics for intelligent alerts and normalization of data across vendor-proprietary metric definitions
- Generate analytics for ingestion by external business systems in support of cost and/or performance optimization e.g. peering and IP transit traffic patterns

Security

- NetSpyGlass supports SNMP v3 for secure device data access and retrieval.
- Information exchanged between NSG Agents and NSG core monitoring servers is encrypted with TLS on the wire.
- Additionally, NSG Agents authenticate to NSG core monitoring servers using certificate and key pairs.
- NSG supports OpenID protocol for operator/user authentication and supports integration with Google Accounts, Azure Active Directory and Ping Federation.

DEVICES	VENDORS
Firewalls	CISCO, JUNIPER NETWORKS
Load Balancers	A10 Networks, f5
Power Distribution Units	Server Technology, APC
SD WAN	viptela, CISCO
Switches & Routers	ARISTA, BROCADE, CISCO, DELL, hp Networking, JUNIPER NETWORKS, NETGEAR, FORCE10
Switches (IP Fabrics)	BROCADE
Switches (Multi-Vendor)	cumulus networks
Servers	Net-SNMP
Virtualized Routers	VYATTA

System Requirements

NetSpyGlass is a cloud-based (Azure & AWS) Software-as-a- Service (SaaS) offering. There are no up-front investments in hardware or software to start monitoring networks of any size.

NetSpyGlass does require installation of Linux-based "Agents" on the user's network. Agents can be hosted on a virtual machine (VM) or a shared machine and do not require dedicated hardware.

Typically, Agents should be deployed in close proximity to the devices they will monitor to minimise potential effects of network latency.

Getting Started

NetSpyGlass was built by network operators for network operators who use the service in large-scale production networks.

With NetSpyGlass, network operations teams now have a consolidated toolset with which to reduce costs caused by network downtime while increasing performance through granular visibility, more effective workflows, analytical insights and planning support.

Visit our website try our Demo or contact us to discuss your network monitoring challenges - here.

NetSpyGlass

About Happy Gears Inc.

Happy Gears provides comprehensive network performance monitoring, diagnostics and topology visualization solutions for the multi-vendor data center and WAN networks. The Company's first product – the NetSpyGlass platform was built for NetOps, DevOps, Security and Application teams to have real-time dashboards and histori-cal network topology maps along with the ability to search and overlay network-related information (i.e. faults, alerts, and performance data). Happy Gears customers include Enterprises, SaaS Operators, Data Centers and Service Providers struggling to conquer the complexity of modern web-scale networks.

For more information visit **https://www.netspyglass.com**