





splunk>



Delivering science and technology
to protect our nation
and promote world stability

Cluster Monitoring Basics

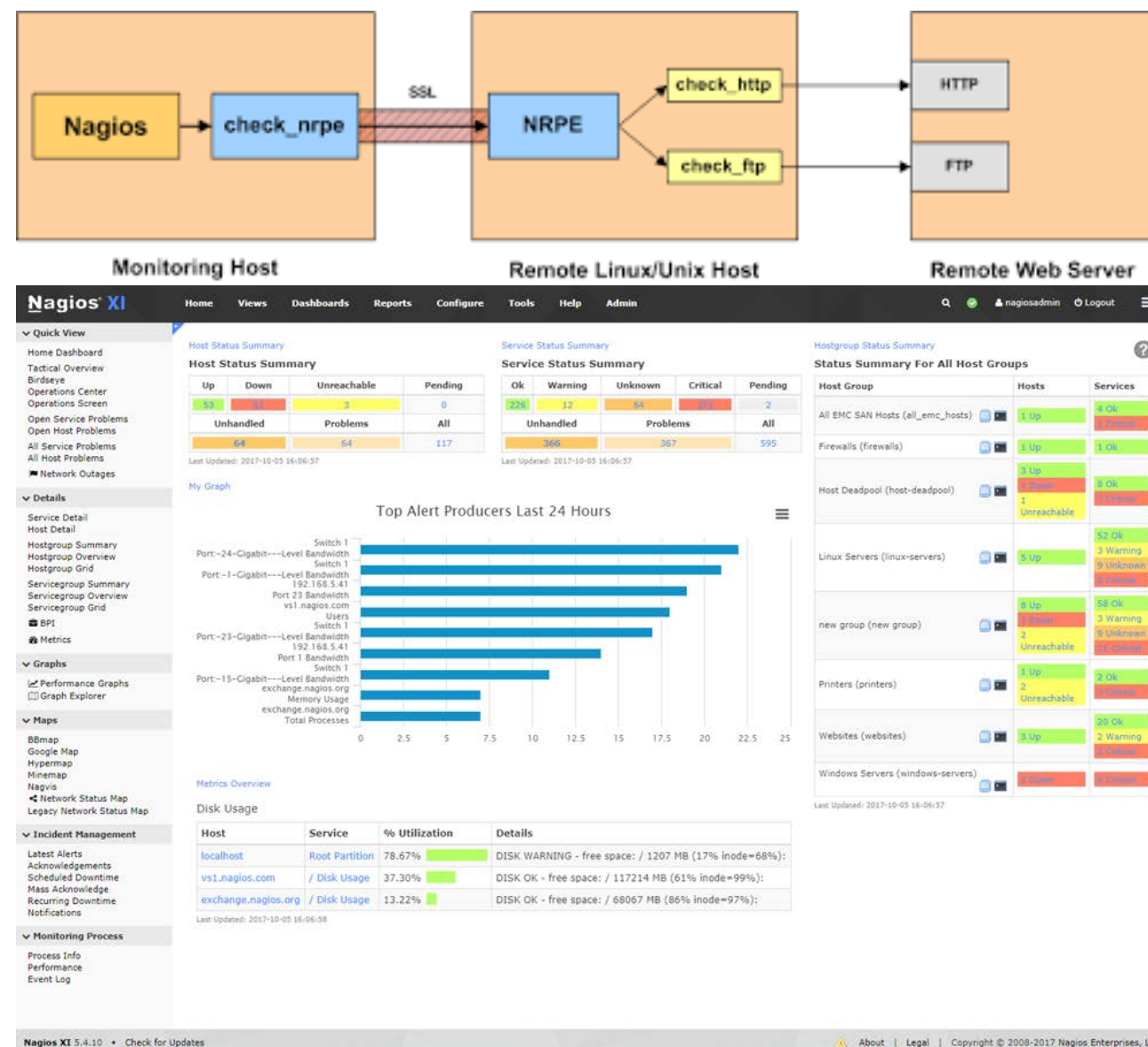
Presented by CSCNSI

Why monitor clusters?

- Detect failures in the system:
 - Did we get a transient failure that might have affected a job?
 - Did a service fail on our master?
 - Did we get a hard failure, like a hardware issue?
 - Are we getting consistent failures, like a design flaw?
- Learn about load and utilization:
 - How hard is our cluster working?
 - How balanced is our workload?
 - Who is using the most of what?
 - Are we consistently under CPU pressure? Memory pressure?...
 - Help to inform future design choices.

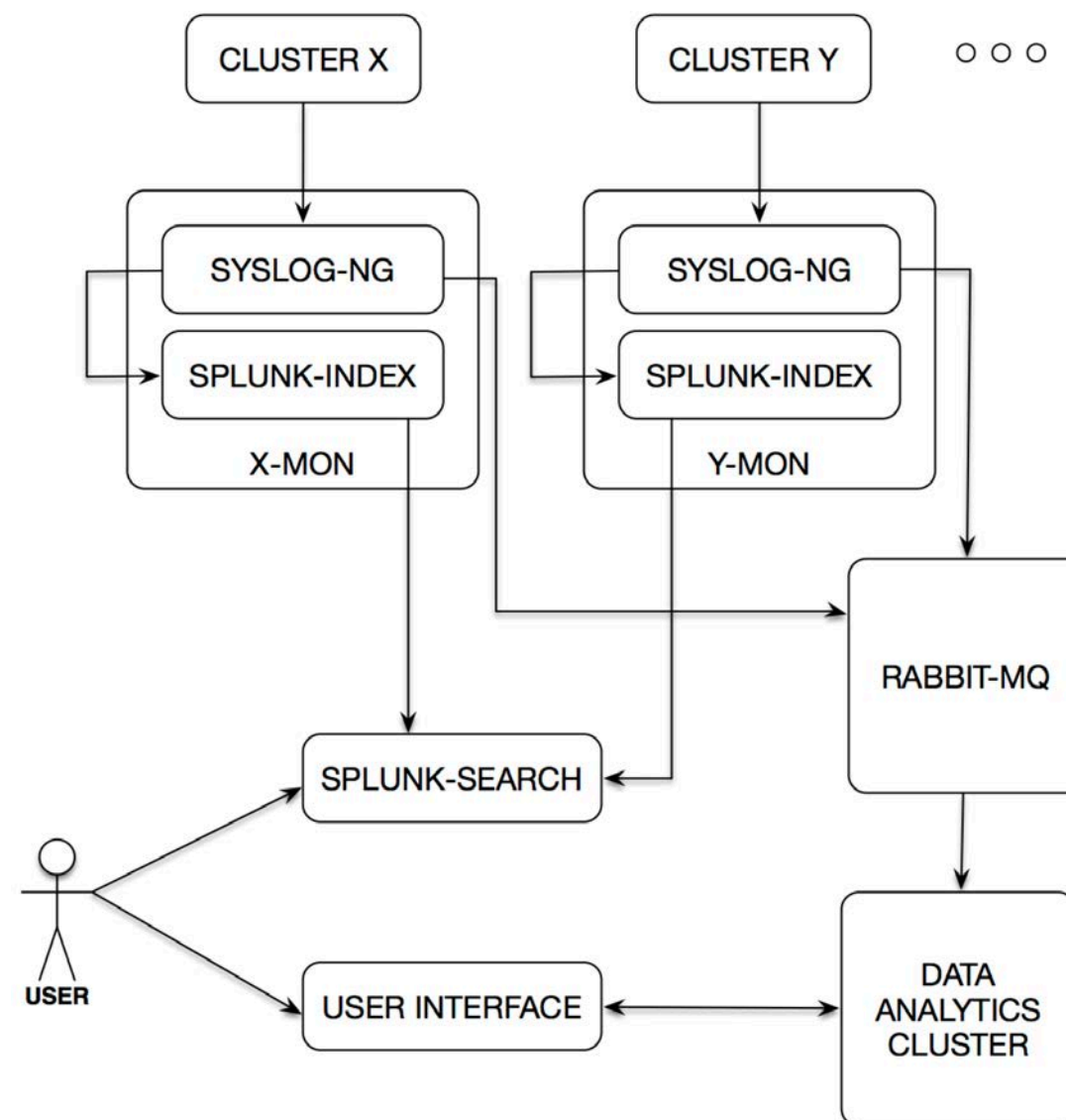
Active monitoring

- Used for:
 - Active polling for service states
 - Functionality checks
 - E.g. try to grab a VNFS image; does it succeed?
- Good for:
 - Getting early indication of failure
 - Things that naturally fit a “pull” model
- Tools that do this:
 - Nagios
 - Zabbix



Passive monitoring

- Used for:
 - Collecting data that already exists
 - System logs
 - Telemetry data, e.g. power utilization, system load...
 - Collecting system usage info
 - Collecting system errors
- Good for:
 - Passive alerts of system failure
 - Tree-like scaling
- Tools that do this:
 - LDMS
 - Splunk
 - rsyslog



Monitoring Infrastructure: The Challenges of Moving Beyond Petascale. – A. Bonnie, LANL

Scaling & performance concerns

- Latency
 - The bigger we grow and more we “fan-out” our monitoring, the more delay before we get a message
 - If we want to actively respond to events, we need to keep latency low
- Overhead
 - Any monitoring task will create a burden on the CPU &
- Jitter
 - Work on the network, CPU, etc. can cause brief interruptions for research applications
 - Routine interruptions can lead to significant over-all loss in performance
- Moving “out-of-band” to reduce Overhead & Jitter
 - We can keep monitoring network traffic to dedicated networks
 - We can even get some data through the BMC instead of the CPU

Active vs. Passive Monitoring

- Pro
 - Active detection of failure
 - Arbitrary scripted checks
 - Con
 - Does not “fan-out” well
 - Can cause a lot of network and CPU traffic (high overhead & jitter)
- Pro
 - Great for collecting telemetry data
 - Tools like RabbitMQ/LDMS support fanning out to very large scale
 - Can be coordinated out-of-band to reduce jitter
 - Con
 - Can produce huge amounts of uninteresting data
 - Alerts and analytics produced post-collection

Questions?