HP Server Automation: Remediation Performance and Scalability Improvements Release Summary Report

Product	HP Server Automation (SA)
Release	9.05 Release
Document ID	SA_Remediation_Rearchitecture-v1-0.doc
Last Updated	October 28, 2011

Quick Links

Overview

HP Sever Automation ("Opsware") was architected to address the challenges of providing a unified world-view of multiple datacenters and distributed targets, spanning security boundaries. Connections to managed servers, either within the datacenter or at remotely distributed sites, are maintained through open, neutral protocols (such as HTTP/HTTPS) that can cross firewalls. Most management operations are built on top of these open standards, allowing interoperability with existing installations and easy extensibility. One of the features of Server Automation is that it is scalable to the largest enterprise networks, in that it efficiently delegates work out to the individual managed nodes, while maintaining one central world view and one reporting structure of the entire enterprise.

IT Administrators are required to manage all of their platforms efficiently, respond quickly to new requirements, and provide audited reports showing their compliance to operational requirements. The IT management solution may be required to perform rapid, guaranteed and audited delivery of content to the distributed targets. For example, a newly-uncovered security issue exposed on a widely-adopted operating system platform (Linux, Windows, other) may require IT managers to perform immediate remediation to tens of thousands of targets, and to show compliance to the required operational standards.

Server Automation can perform digital content delivery across many thousands of managed servers. The digital content can be any form of file or content copy that is to be delivered to the target: new Application delivery (such as that provided by ADM functionality), OS Patching content (e.g. Windows "Patch Tuesday" bundles, RHN Monthly Errata, Security hot fixes, etc.), customer package content (e.g. Webserver or Application Server content, IT/Administrator content, custom package delivery). The Remediation feature of Server Automation allows efficient, scalable, and audited delivery and installation of any of these forms of content to the managed server targets. For one recent Server Automation Use Case requirement, the provisioning of a new datacenter required the delivery and installation of 5 Gigabytes of custom content to each of 1000 target servers in the datacenter, all to occur within an allowed 1 hour maintenance window.

SA Remediation uses neutral and open HTTP/HTTPS protocols and simple http file transfers to deliver content to each target. This traffic can be proxied through gateways and across security boundaries. In earlier versions of SA, Remediation content delivery was implemented with replicated webservers. Major architectural improvements have been made to this feature to allow efficient delivery when scaled to deliver content to many thousands of targets. The new delivery mechanisms of Server Automation can provide content delivery at nearly full wire speed (nearly full utilization of 1 Gbps and 10 Gbps networks are demonstrated in the lab setting, real world performance will depend on the particular environment in which it is deployed). This new delivery mechanism requires no changes to existing customer software policies or operational procedures, and can be upgraded into existing Opsware installations already deployed in large datacenters.

This paper provides a case study on improved content delivery for SA Release 9.05 in the context of a common SA job type, Remediation of Software Policy digital content.

Executive Summary

In a direct comparison of SA Remediation performance from the 9.04 to 9.05 Releases, SA Remediation performance is shown to be improved by factors ranging from 4x to 9x, depending on the size and scale of the Remediation content.

SA Remediation functionality exhibits reasonably smooth and predictable throughout under these test conditions. SA operations may be extrapolated from the overall guidance provided.

Job distribution is balanced across the 2 Slice application servers in the SA Core. The SA Word (content delivery webserver) and Way (workflow manager) components balance the workload. The overall operation scales out efficiently across the two Slice Application servers.

System resource metrics (CPU, memory, and network resources) show SA internal operation to make efficient use of system resources. Network usage approaches full consumption of the 1 Gbps network link during heavy phases of the Download stage of the operation. [1]

[1] SA Remediation performance has also been characterized operating with 10 Gbps network links in the Performance Laboratory. SA Remediation operates efficiently at large scale on this infrastructure as well, and can serve Remediation content nearly to the full network bandwidth of the 10 Gbps network link. These results are not a subject of this performance whitepaper.

Performance Results

The stated test case was exercised against both 9.04 GA and 9.05 GA [i] Server Automation cores. The SA Core configuration was one Truth database server and two Slice Servers (Word, Way, Twist, Hub, and Gateway components). The managed servers were varied from 1 to 96 targets for different test runs; target Operating System was RedHat Enterprise Linux 5 Advanced Server Update 4, x86-64 version. Further details are given in the appendices to this report.

Software delivery content was in the form of custom software policies that delivered varying numbers of Linux RPM installation packages to each target. The delivery and installation characteristics vary depending on the size and number of installation packages, so guidance is provided along those lines. The following software policies are used in the test:

Remediation Content Delivery Use Cases		
Software Policy	**# RPM packages**	**Total content delivered / target**
1Mx100	100	100 MBytes
1Mx300	300	300 MBytes
1Mx500	500	500 MBytes
1Mx1000	1000	1 GByte

Table 1: Remediation Software Policies (Delivery Content)

The following performance results are demonstrated

Servers Remediated / Minute							
Software Policy	***#Targets***						
	1	***10***	***20***	***40***	***60***	***80***	***96***
1Mx100	0.35	2.09	2.81	3.63	3.29	3.57	3.58
1Mx300	0.15	1.14	1.41	1.56	1.62		
1Mx500	0.09	0.36	0.36	0.40	0.31		

Table 2: Remediation throughput, 9.04 GA – Release without Architecture Improvements

Servers Remediated / Minute							
Software Policy	***#Targets***						
	1	***10***	***20***	***40***	***60***	***80***	***96***
1Mx100	0.30	3.14	6.63	10.91	13.74	15.53	14.40
1Mx300	0.14	1.27	2.95	4.67	5.56	5.45	5.77
1Mx500	0.08	0.83	1.81	2.32	2.90	3.34	3.16
1Mx1000	0.05	0.43	0.80	1.18	1.42	1.60	

Table 3: Remediation throughput, 9.05 GA – Release with Architecture Improvements

For these tests, throughput is calculated by dividing the overall SA Job completion time by the number of managed servers in the Remediation job submitted for processing. This number is most useful to IT administrators and SA architects to use for operational and sizing purposes.

The following graphs demonstrate the improved throughput of the SA Remediation functionality.

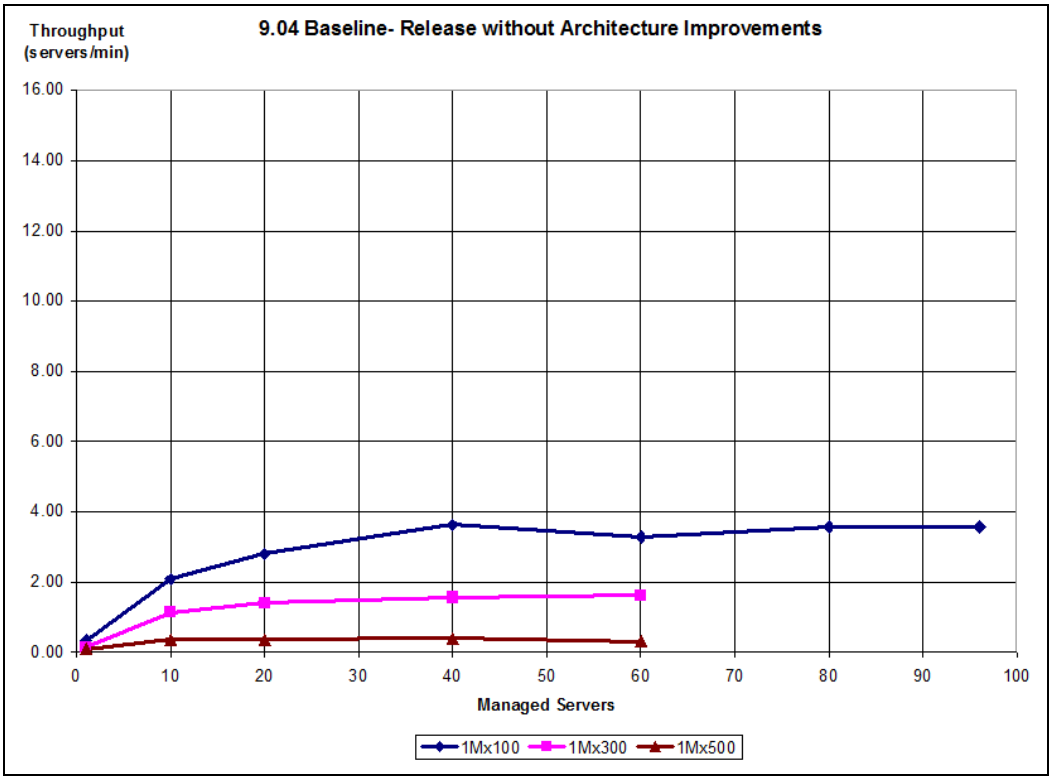

Figure 4: Remediation Throughput: 9.04 GA

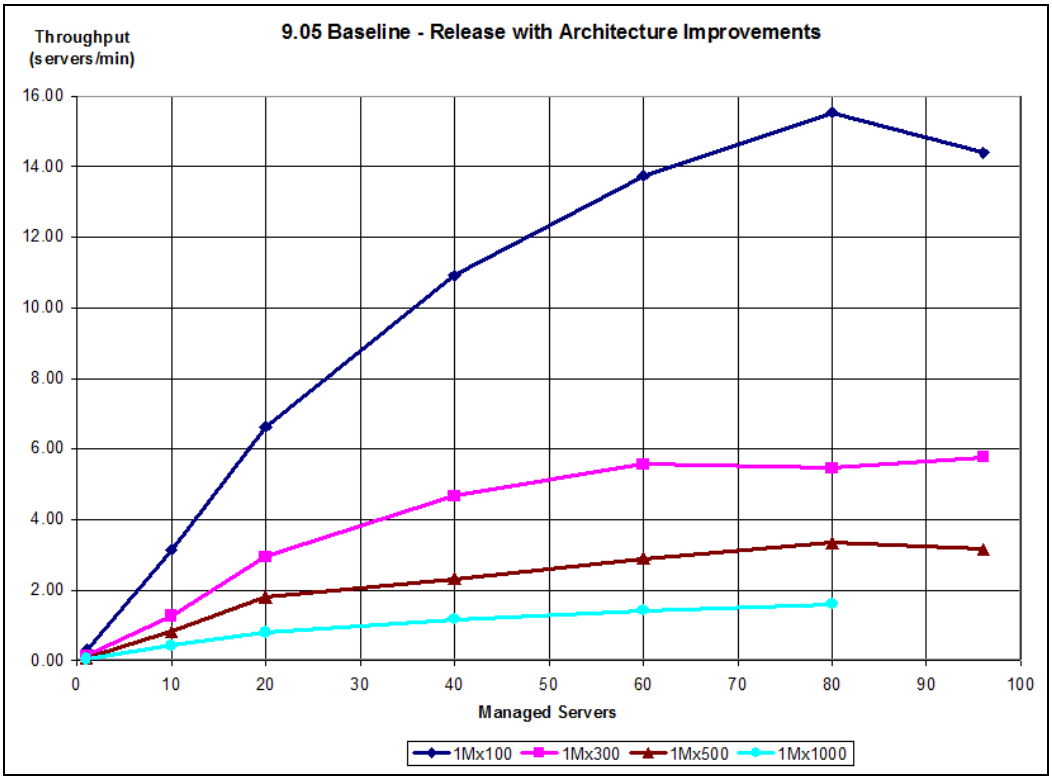

Table 5: Remediation Throughput: 9.05 GA

Use Case Description

The following parameters are used for this use case characterization:

1. Software policies are constructed to deliver and install RPM packages to Linux targets.
2. Policies of varying number or packages and total content size are characterized.

The characterized Remediation policy (test case) does the following:

1. Analyze the existing managed server model to determine applicability of the content. For this performance characterization, all packages are always delivered and installed.
2. Stage content through the SA Agent running on each target.
3. SA Agent pulls delivery packages from the SA Word content store.
4. Packages are unpacked and installed on the target
5. Delivered content is checked for compliance after the installation.
6. Job status is reported back to SA Core.

The test case is programmatically invoked using the SA UAPI through the pytwist interface.

SA Job Characteristics

The following series of graphs show details of the Remediation of 500 MB of content (Policy of 500x 1 MB RPM packages) across 80 target managed servers.

Server Automation efficiently decomposes User Jobs into a number of sub-tasks. SA Core sub-tasks can be distributed across the SA Core servers, and delegates work to the target managed servers while the SA Core maintains management and visibility throughout the job cycle.

For the Remediation job, the SA components of Way (workflow management) and Word (media server) are heavily used and their workload is distributed across the 2 Slice servers. The following graphs of system workflow and of system resource consumption show a good distribution across the SA Core Slice 1 and Slice 2 servers.

In the first plot showing SA workthreads for the job, it is seen that all 80 target managed servers are processed concurrently in Remediation. (See later section on SA Concurrency Settings for additional information). The actual work of doing the job is divided between SA Core-side operations (such as job setup, transmit download, job management, job reporting) and Managed Server Client-side operations (such as receive download, unpack, install content, compliance check). Details of this workflow can be seen in the following graphs.

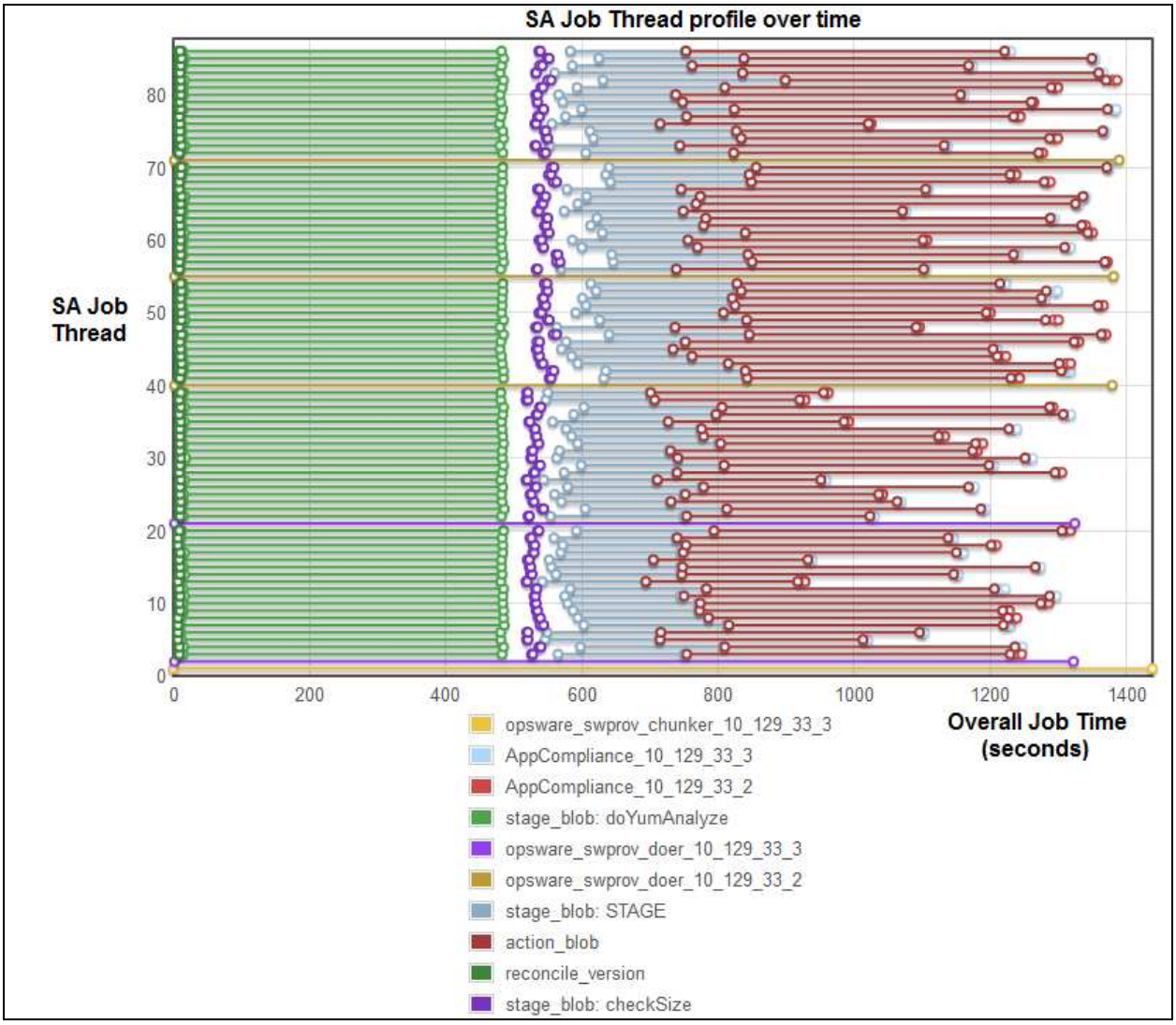

Figure 1: SA Job workthread decomposition: 500 MB content delivered to 80 targets

Server Resource Utilization

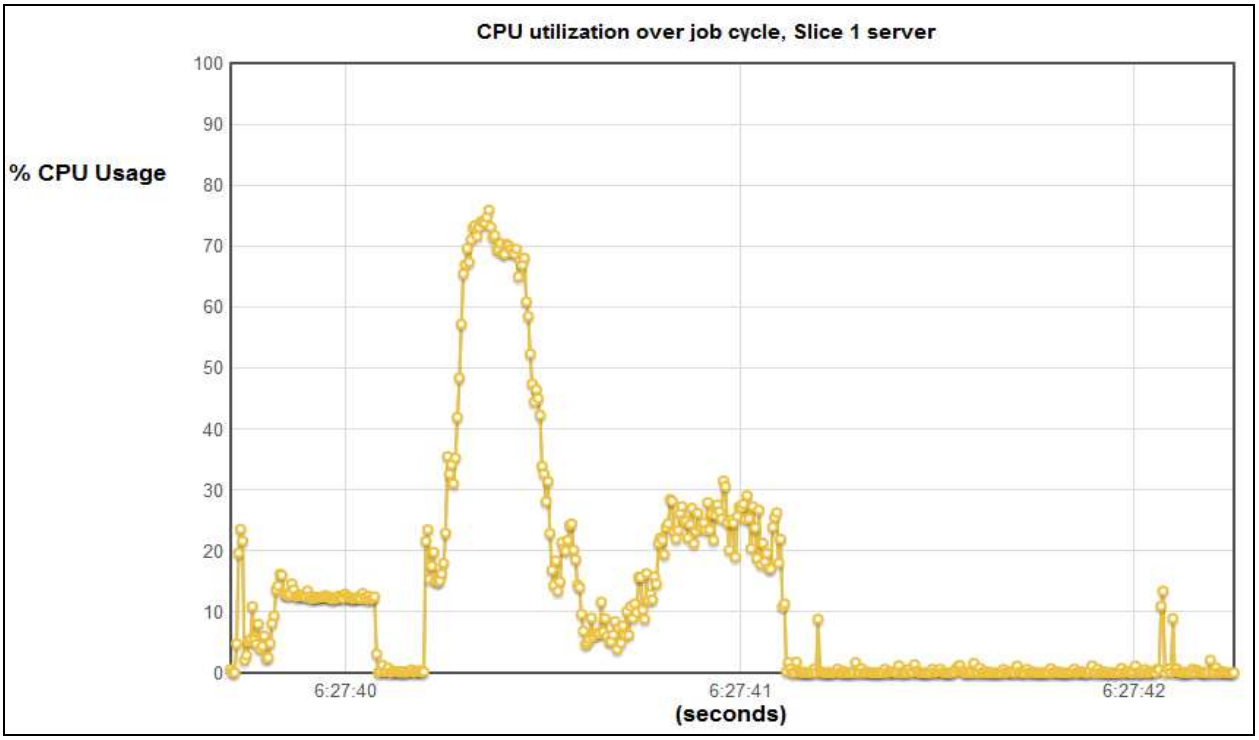

Figure 2: CPU utilization usage over job, Slice 1 server

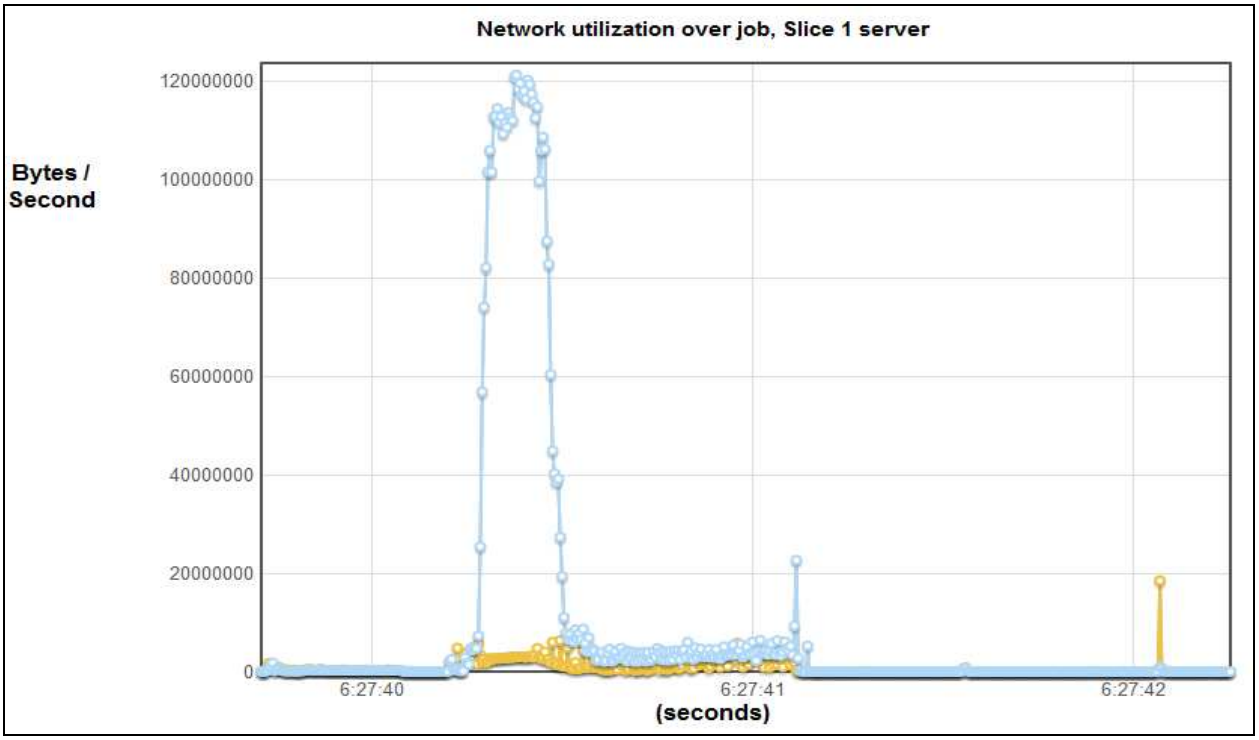

Figure 3: Network utilization over job, Slice 1 server

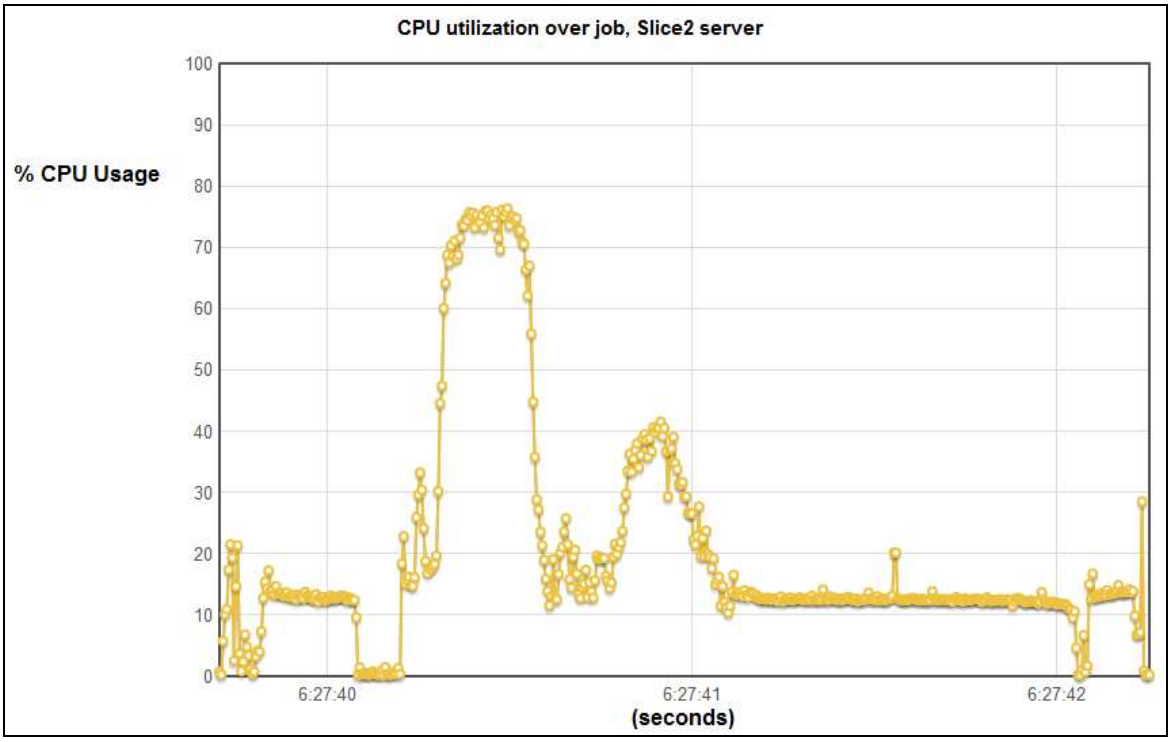

Figure 4: CPU utilization over job, slice 2 server

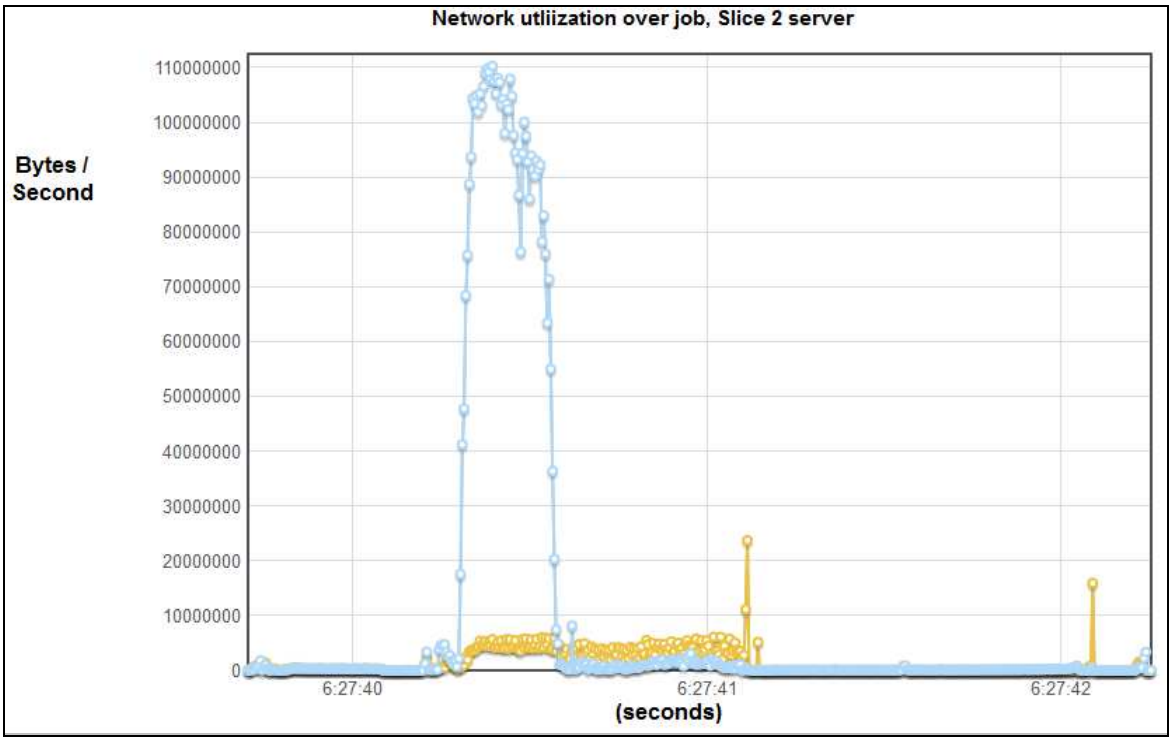

Figure 5: Network utilization over job, Slice 2 server

Server Resource Utilization – details by server function

Slice Servers

In Remediation, content is delivered across the network during the Download phase of the operation. The plots how that SA can serve content at near wire-speed of the Gbps network links, showing very good utilization of resources during the transfer.

Following the content delivery, the SA Agent on the client directs the unpacking, installation, and compliance checking of the digital content. During this phase, CPU and network load on SA Core is light, as most of the work is being done on the managed server during this part of the workflow.

Database Server

SA utilization of the Truth database server is show in the following graphs. Database CPU usage increases during SA-Core side operations, and is low during Client-side target operations. Database server network utilization stays relatively lower throughout the operation (network graph scale is 1/10 that for the Slice network graphs).

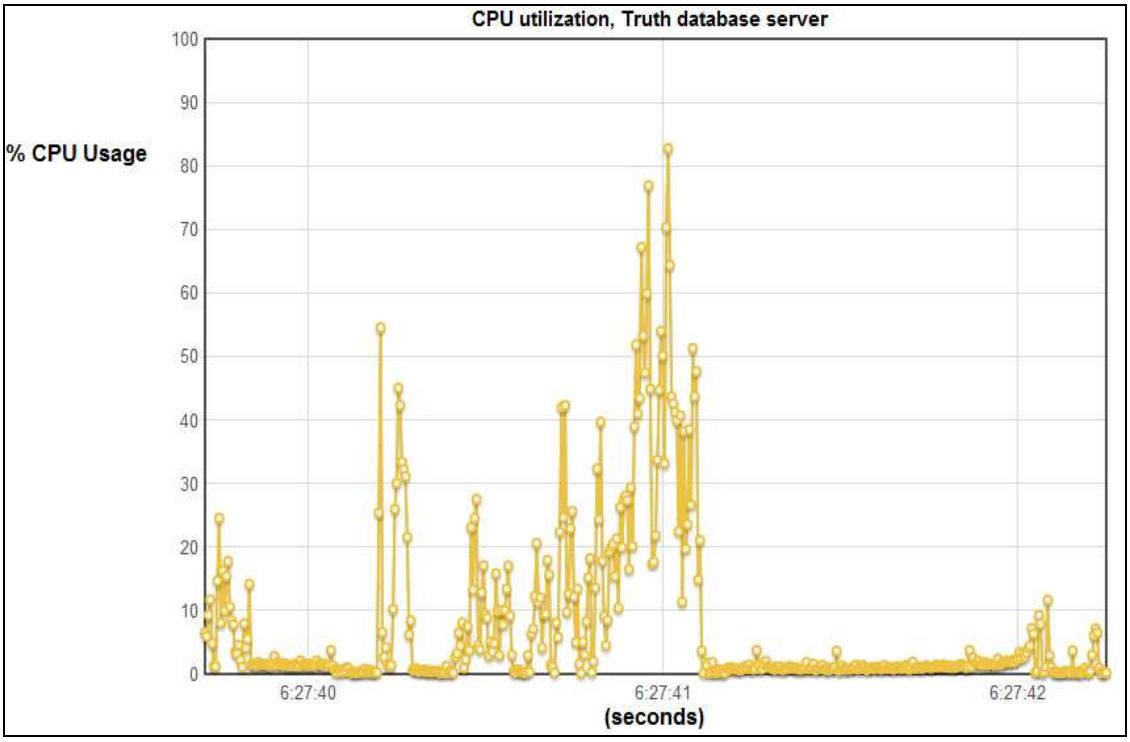

Figure 6: CPU utilization during job workflow, Truth Database server

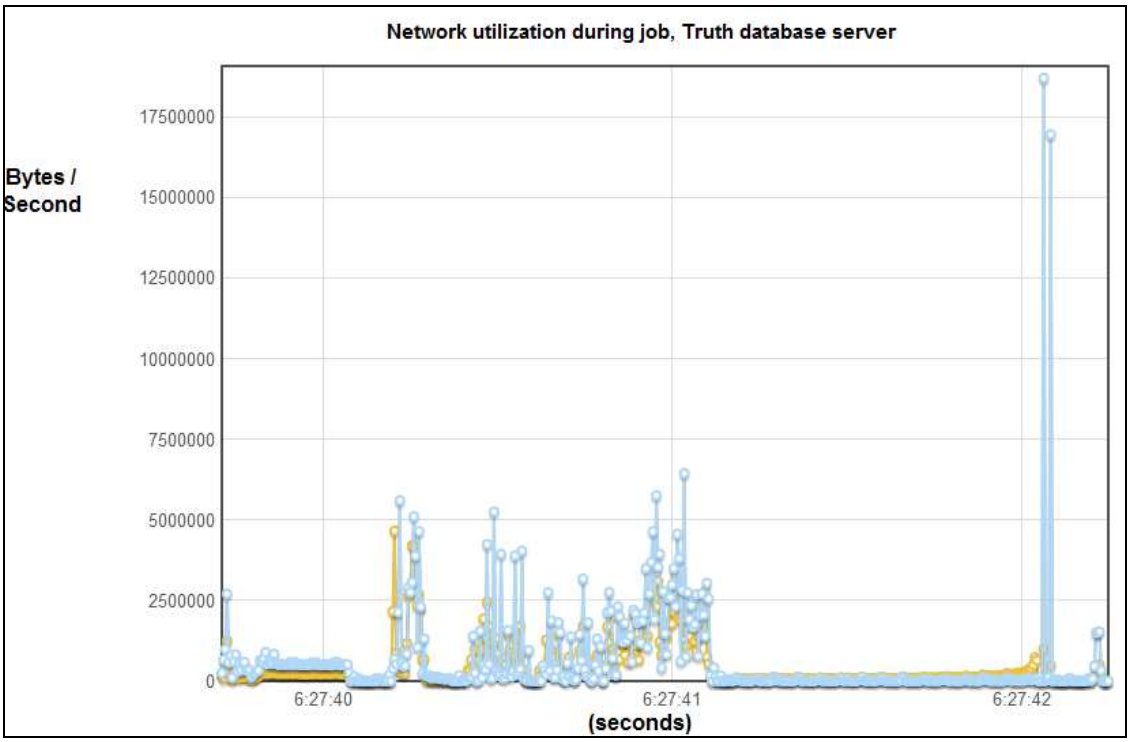

Figure 7: Network utilization during job, Truth database server

Remediation Job Concurrency and Job Decomposition

Remediation job concurrency is controlled by the "**way.max_remediations**" and "**way.max_remediations.action**" configurable parameters. **way.max_remediations** is the total number of remediation actions allowed concurrently per Slice server, in the earlier stages of the operation (where most of the work is being done on the SA Core side) on each Slice server in the SA Core. **way.max_remediations.action** is the number of remediation operations in the "Action" phase (where most of the work is being done on the Client side) of the operation.

These parameters may be varied if the expected workload consists of well-behaved Job submission of particular characteristics. The setting of this parameter should be based on the resource demand characteristics of the Remediation job that is being performed. Configurable parameters are the settings of finite resources within SA Core, and should be understood if they are changed.

This configurable parameter can be adjusted using the Server Automation Web Client interface, at the following path:

OCC web>>Administration>>System Configuration>>Command Engine>>way.max_remediations

OCC web>>Administration>>System Configuration>>Command Engine>>way.max_remediations.action

For these series of tests, these parameters were left at their default value of 50 and 100 respectively. Overall SA Core concurrency is this number multiplied by the number of Slice instances in the Core. The following table gives the settings of this parameter for each configuration of the number of Slice instances.

Parameter	max_max_remediations (per Slice server)	max_max_remediations.action (per Slice server)
Default value	50	100

Table 6: Concurrent Remediation operations over the number of Slice Servers

Conclusions

For the stated configurations of the system under test, the SA Remediation feature performance has been improved by factors ranging from 4x to 9x for the tested use cases. The effects of varying numbers of install packages and total payload are characterized. SA operation is stable and predictable as the number of managed servers is increased. System resource metrics show SA operation makes efficient use of system resources.

Hardware Configuration

SA Core Servers

Server Role	"Truth" Database
Hardware Specs	Local Disk: 2x 72GB 10K SA RAID-1 (Linux boot) Local Array: MSA50 12x 72GB 10K SAA, Ultra-SCSI-3 connect Memory: 32GB PC5300 OS: RHEL 5 AS 64-bit CPU: 2x Quad-core 2.66 GHz. Intel Xeon 5150 Model: HP Proliant DL360G5
Network Config	Network: 1 Gbps LAN, quiet VLAN
Software Specs	Oracle 11.10.2.0 Standard Edition
Server Role	Infrastructure plus Slice 1 services Data access engine (Spin - primary) Distributed Command Engine (Way) Web service API (Twist) Opsware Global File system (Hub) Spoke Word Master Gateway and Agent Gateways Media Repository (NFS,SMB)
Hardware Specs	Local Disk: 2x 72GB 10K SA RAID-1 (Linux boot, SA installed) SAN Attach: 2Gbps dual path FC, MSA1500 Array (Media store) Memory: 16GB PC2-5300 OS: RHEL 5 AS 4 64-bit CPU: 2x Quad-Core 2.66 GHz Intel Xeon 5355 Model: HP BL460cG1
Network Config	Network: 1 Gbps LAN, quiet VLAN
Software Specs	Server Automation SA 9.04 GA Build 3779 Server Automation SA 9.05 GA Build 4108 plus Wayscript Hot fix.
Server Role	Slice 2 services Spin – secondary Distributed Command Engine (Way) Web service API (Twist) Opsware Global File system (Hub) Spoke Word Agent Gateways
Hardware Specs	Local Disk: 2x 72GB 10K SA RAID-1 (Linux boot, SA installed) SAN Attach: none Memory: 16GB PC2-5300 OS: RHEL 5 AS 4 64-bit CPU: 2x Quad-Core 2.66 GHz Intel Xeon 5355 Model: HP BL460cG1
Network Config	Network: 1 Gbps LAN, quiet VLAN
Software Specs	Server Automation SA 9.04 GA Build 3779 Server Automation SA 9.05 GA Build 4108 plus Wayscript Hot fix.

Table 7: SA Core System Configuration

Hosts and Managed Servers

Server Types	For Test cases of Remediation: ▪ HP Blade servers, hosting VMware ESX 4.0 Virtual Machines
Hardware Specs	Local Disk: 2x 72GB 10K SA RAID-1 (ESX boot) SAN Attach: 4Gbps dual path FC, EVA4400 Array (VM images) Memory: 32GB PC2-5300 OS: VMware ESX Server 4.0 CPU: 2x Quad-Core 2.66 GHz Intel Xeon 5355 Model: HP BL460c
Managed Server Specs	RedHat Linux 5 AS x86-64 SA Agent Version: 40.0.0.1.122-linux-4AS-X86_64 1x Virtual CPUs 4 GB memory 40 GB disk RAM disk filesystem as target for download and install directories. This circumvents performance issues caused by the virtualized filesystem environment.
Configuration Details	▪ Managed Servers are distributed across up to 12 physical Blade Servers for each test. ▪ All device groups will be distributed as evenly as possible across the Vmware VM servers during the runs.
Network	Network: 1 Gbps LAN, 30 Gbps Cisco network Switch, quiet VLAN
Additional Notes	

Table 8: Test System Managed Server Configuration

Test Environment

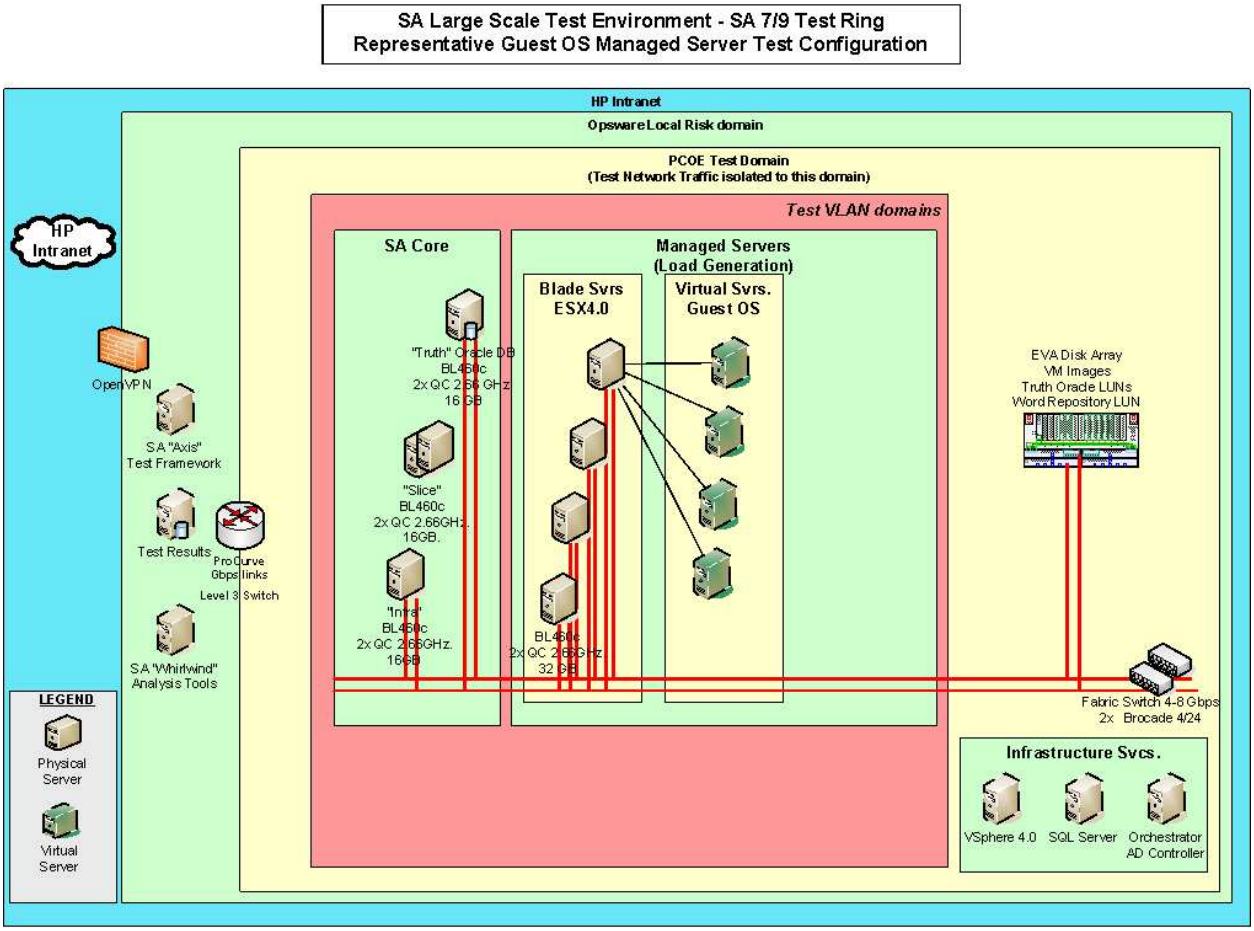

Figure 8: SA Core and Managed Servers Test System