

CommuniGate Pro SIP Performance Test on IBM System z9

Technical Summary Report Version V03

Version : 03
Status : final

Updated : 16 March 2007

1 Executive summary

In this document, you'll find the results of tests run in the PSSC Montpellier in July – September, 2006.

The request coming from IBM Brazil was to execute a performance test of the CommuniGate Pro Internet Communications Platform in the System z9 environment to meet a customer request for information on scaling CommuniGate Pro on System z.

The main objectives of the performance test were:

- To prove the z9 platform as a suitable platform for CommuniGate Pro in both “single-server” and “SIP Farm Dynamic Cluster” architectures.
- To prove the capability of the z9 platform for Service and Telecommunications Providers, with a subscriber base of over 20 million accounts.
- To validate the scalability of a VoIP service offering delivered on the CommuniGate Pro SIP Farm Dynamic Cluster running on System z Linux.
- To demonstrate the administration, reliability, and scalability benefits and cost-effectiveness of running a CommuniGate Pro SIP Farm Dynamic Cluster on the z9 platform.

During this performance test the following objectives were achieved:

- The z9 platform proved to be a superb platform for the CommuniGate Pro Internet Communications Platform, in terms of overall performance as well as scalability. Ultimately, the testing demonstrated a subscriber base of 25 million accounts (all of which were enabled not only for SIP/RTP-based VoIP but also e-mail and calendaring) with 5 million accounts registered simultaneously.
- The z9 platform demonstrated very good overall vertical scalability as the number of Linux virtual machines was increased to a tested maximum of 20 z/VM Linux virtual machines for the CommuniGate Pro SIP Farm Dynamic Cluster (with another 10 z/VM Linux virtual machines used for other performance test needs such as an NFS Server, DNS servers, and load-generators), with excellent CPU usage rates and near-linear performance increases as the number of virtual CPUs was increased.
- The service and clustering capabilities of CommuniGate Pro in conjunction with the reliability and efficiency of the z9 platform demonstrates a “model architecture” that Service and Telco Providers should be strongly considering to satisfy both current VoIP subscriber levels and future growth of VoIP services.

2 Introduction

CommuniGate Systems develops carrier class Internet Communications software for broadband and mobile service providers, enterprises and OEM partners worldwide. Their flagship product - the CommuniGate Pro Internet Communications Platform is a recognized leader in scalable e-mail messaging, collaboration, and multi-media over IP (VoIP), running on more than 30 major computer platforms. Its unsurpassed scalability and feature set have won more industry awards than any other software communications solution on the market.

Session Initiation Protocol (SIP) is an application-layer control (signaling) protocol for creating, modifying, and terminating sessions with one or more participants. These sessions include Internet telephone calls, multimedia distribution, and multimedia conferences.

3 Performance test preparation

3.1 Technical infrastructure

3.1.1 Hardware

- IBM System z9 109 S38 (S54 starting 10th August):
 - 38 (54) processors
 - 128 (256) GB of memory
 - 8 x FICON channels (Ficon Express Adapter)
 - 2 x OSA Express Fast Ethernet Adapter
 - 2 x OSA Express2 10 Gigabit Ethernet
 - IBM TotalStorage DS8100
 - IBM TotalStorage Enterprise Tape Drive 3590 tape drive for backup purpose

3.1.2 Software

3.1.2.1 z/VM Virtualization

- Operating system: IBM z/VM 5.2 RSU 5202 (March 2006)

3.1.2.2 Operating System on Servers

- Operating system: SUSE Linux Enterprise Server Version 9 Service Pack 3, 64bit

3.1.2.3 Performance monitoring & reporting tools

- z/VM Performance Toolkit (optional component of z/VM)
- SUSE Linux APPLDATA monitor (built-in component of SUSE Linux Enterprise Server)
- RMFPMS for Linux:
<http://www-03.ibm.com/servers/eserver/zseries/zos/rmf/rmfhtmls/pmweb/pmlin.html>

3.1.2.4 Network tools

- BIND 9 for DNS routing
- Linux Virtual Server (LVS) using the “direct routing” method

3.1.2.5 Test tools

- SIPp for SIP traffic generation

3.1.2.6 Software Under Test

- CommuniGate Pro version 5.0.10 for 64-bit zSeries Linux (“s390x” platform)

3.1.3 Network

Two physical networks were set up:

- Fast Ethernet for the administration network 10.3.48.x. This network was accessible remotely by teams of CommuniGate Systems and IBM Brazil with a public IP address through.
- 10 Gigabit Ethernet for the injection network (direct cable connection) 192.168.1.x

1 virtual network was set up:

- HIPERSOCKETS network 192.168.2.x. This network was used for communication between Linux server for some of the runs

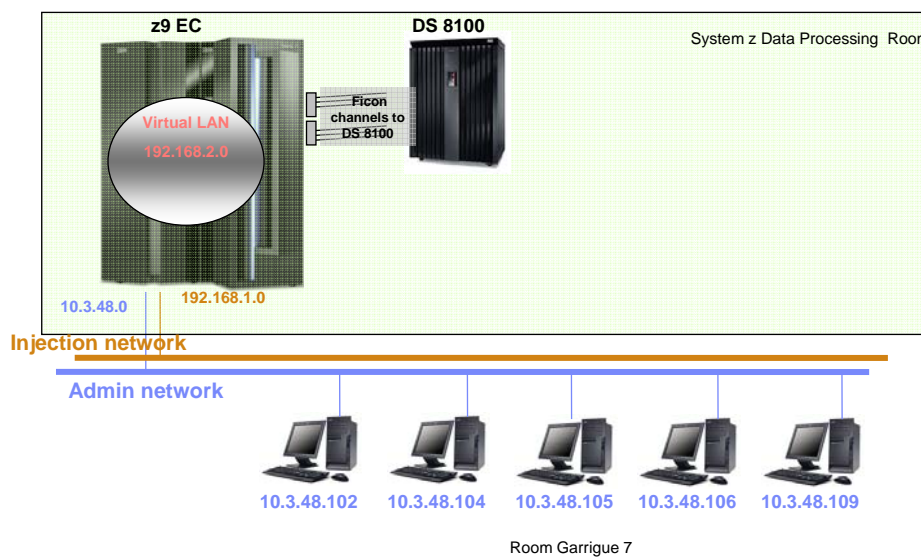
Hipersocket network is a virtual TCPIP network defined on the microcode level of the System z machine. It enables network communication between virtual machines running in different LPARs or under z/VM. Up to 16 Hipersocket networks could be defined in one machine.

The Hipersocket network is a particularly efficient mechanism for “intra-cluster” requirements within the CommuniGate Pro Dynamic Cluster. The Dynamic Cluster architecture behaves in some ways like a grid system, with intra-cluster exchanges between cluster nodes for load distribution, traffic allocation, call and PBX bridging, and application-level redundancy. Within the Dynamic Cluster, all nodes are active, and all accounts are serviced on all nodes in the cluster. There is no segmentation or division of the subscriber base, and one of the ramifications of this great simplicity are intra-cluster communications which benefit greatly from the very low latency and excellent networking performance of the z9.

Europe ATS - PSSC



Benchmark Architecture



© 2007 IBM Corporation

3.1.4 CommuniGate Pro Clustering Architecture

The following section briefly introduces the CommuniGate Pro SIP Farm Dynamic Cluster technology, and describes the logical architecture of the test cluster.

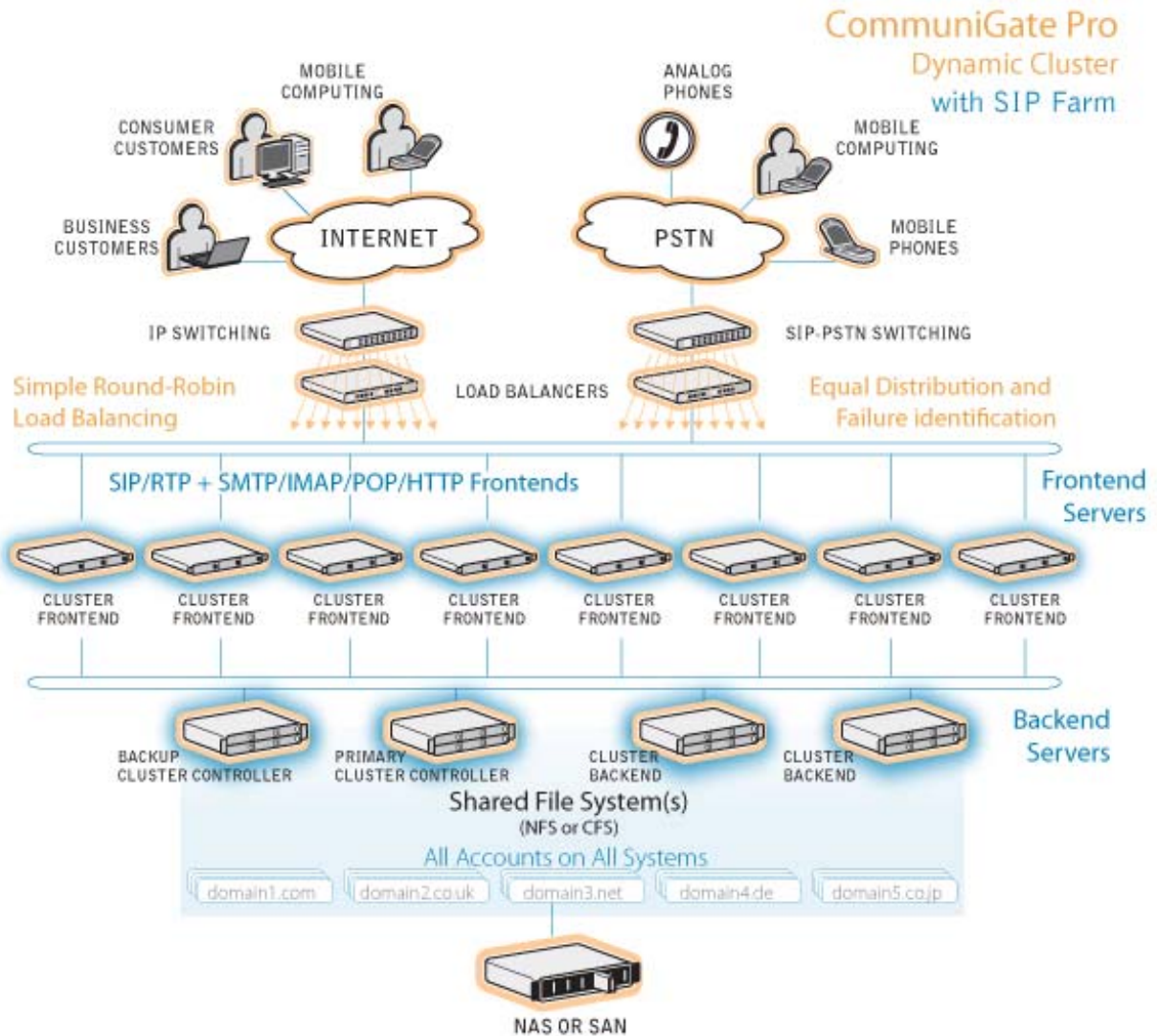
3.1.4.1 Introduction to the Dynamic Cluster and SIP Farm

SIP Farm is CommuniGate Pro's unique technology for clustering voice-over-IP (VoIP) for 99.999% uptime, redundancy, and scalability. Both Dynamic Cluster and Super Cluster deployment implementations can be clustered with SIP Farm, and the members of a cluster allocated to the SIP Farm can be based on traffic or regional geographic node placement.

CommuniGate Pro SIP Farm allows providers to sustain a million busy-hour call attempts and call rates of over 300 calls per second on each cluster frontend member, with a large cluster capable of call rates of tens of millions of Busy-Hour Call Attempts (BHCA), while providing innate resilience in case of system or even site failure. Incoming SIP UDP packets and TCP connections are distributed to Frontend member nodes using regular/simple load balancers. In the event of the gain or loss of a SIP Farm member node (such as for hardware failure or

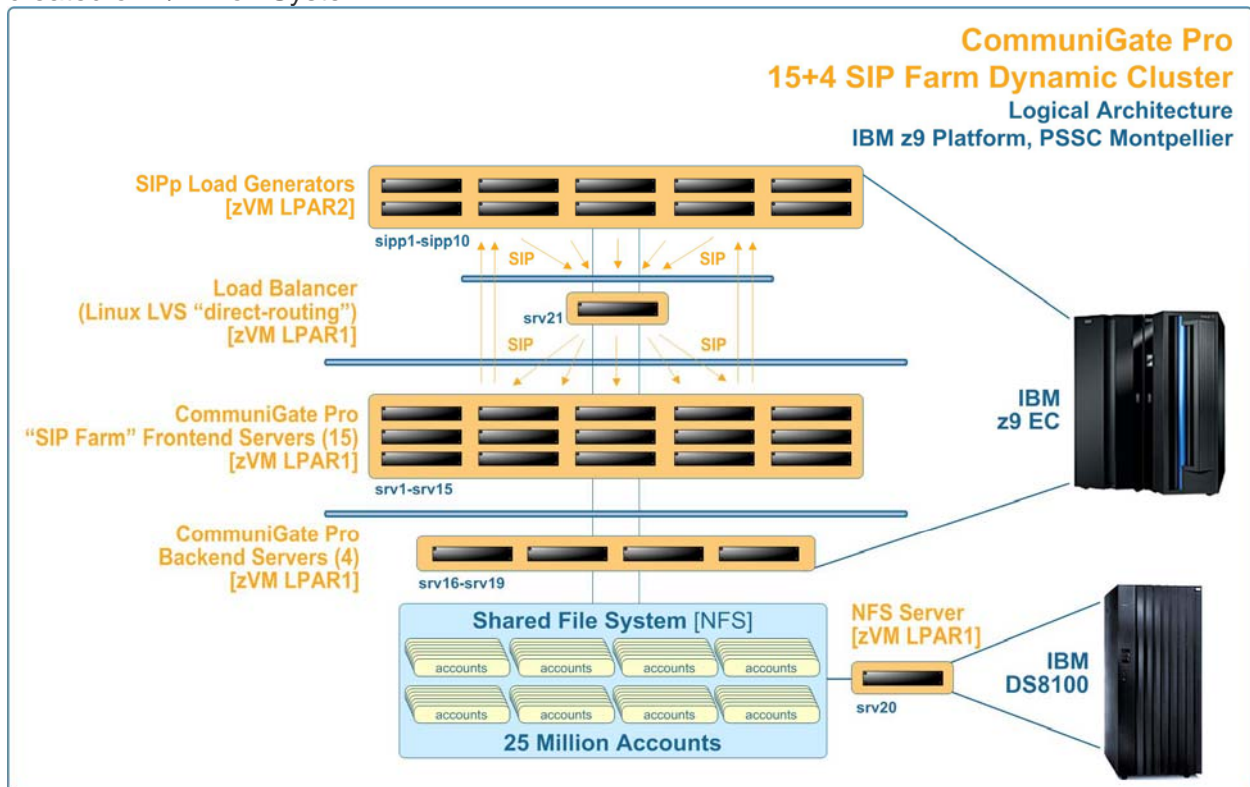
“rolling upgrades”), the traffic is redistributed to other SIP Farm members to maintain consistent signaling.

The following diagram demonstrates a 8+4 Dynamic Cluster [8 Frontends, 4 Backends], architected in a simple layout where all Frontends are part of the SIP Farm, and therefore all Frontends use the same configuration and provide all services to all domains and accounts:



3.1.4.2 Logical Architecture

The following diagram illustrates the logical architecture of the CommuniGate Pro SIP Farm Dynamic Cluster - as well as the storage layout, load balancer, and load generation systems – used in the CommuniGate Pro z9 platform testing. All Linux instances used in the testing were created on z/VM on System z.



3.2 Environment setup

3.2.1 DS8100 configuration

3.2.1.1 Physical Layout

The DS8100 disk storage system was configured with 4 Logical Control Units (LCU) connected using 8 FICON channels to the z9 machine.

Each of these LCUs emulated 80 x 3390 model 3 DASDs, 320 devices with these device addresses:

- 7200 – 724F
- 7300 – 734F
- 7400 – 744F
- 7500 – 754F

3.2.1.2 Disks allocated to z/VM and Linux virtual machines were evenly distributed among all LCUs. Logical Layout

The required “Shared File System” (as discussed previously in the section “Clustering Architecture”) for this testing consisted of a single 255 GB Linux LVM Logical Volume built from 116 3390 model 3 DASDs, attached to srv20 via FICON Channels as full-pack minidisks:

```
srv20# for i in $(cat DASD); do fdasd -a /dev/${i}; done | tee log 2>&1
srv20# for i in $(cat DASD); do pvcreate /dev/${i}1; done | tee log 2>&1
srv20# vgcreate -p 256 -s 32m datavg $(cat DASD)
```

On this Logical Volume, an XFS filesystem was created with a block-size of 4kB:

```
srv20# mkfs.xfs -f -s size=4096 /dev/datavg/data1v1
```

Once active with the CommuniGate Pro Dynamic Cluster, and after all 25 million accounts and account meta-data (e.g., password, real-name) had been enabled on the cluster, this primary data volume “Shared File System” had approximately the following characteristics (the volume usage varies throughout the testing, but not significantly due to the profile of these tests):

1K-blocks	Used	Available	Use%	Mounted on
277348288	229616928	47731360	83%	/cgp

3.2.2 System z9 configuration

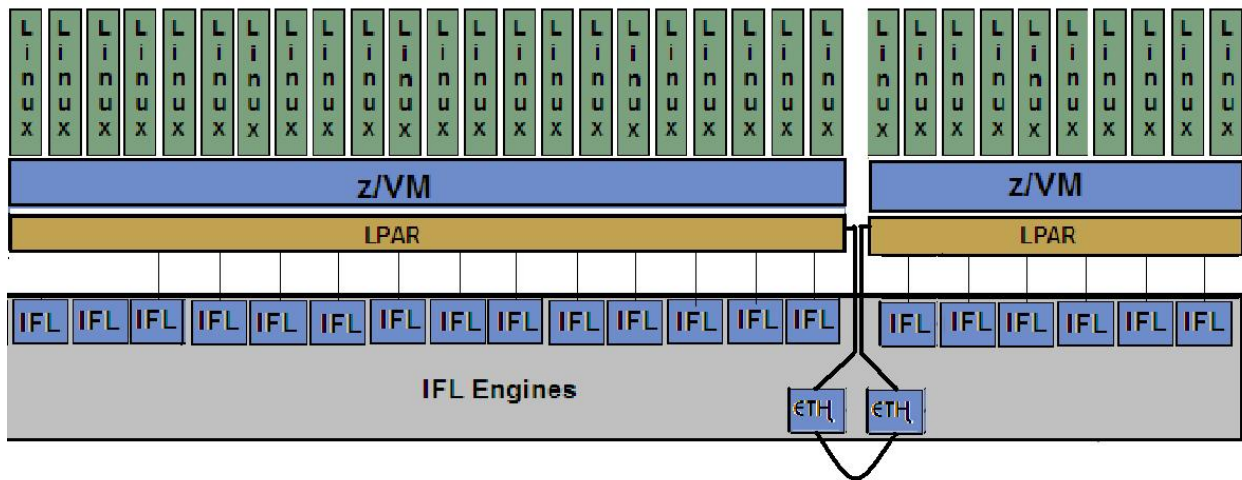
Three logical partitions (LPARs) were defined on the z9 machine out of which two were used during the tests.

3.2.2.1 Logical Partition 1 (LPAR1)

- Number of CPs (Central processors): 12 (24 with S54 model) CPs were dedicated to the LPAR1
- Central Storage: 96 GB Central Storage dedicated to the LPAR1
- Expanded Storage: 32 GB Expanded Storage dedicated to the LPAR1 (in order to improve the performance of the z/VM paging activity)
- Operating system: z/VM 5.2 RSU 5202

3.2.2.2 Logical Partition 2 (LPAR2)

- Number of CPs (Central processors): 6 (24 with S54 model) CPs were dedicated to the LPAR2
- Central Storage: 48 GB Central Storage dedicated to the LPAR2
- Expanded Storage: 16 GB Expanded Storage dedicated to the LPAR2 (in order to improve the performance of the z/VM paging activity)
- Operating system: z/VM 5.2 RSU 5202



4 Test Scenarios and Summary Results

4.1 Table of Scenarios and Results

Test Title	Test Summary	CommuniGate Pro Architecture	Load Balancer used? (Yes/No)	NAS Server used? (Yes/No)	REGISTER/auth used? (Yes/No)	Calls per second (as measured by sipp, 60-second call duration)
R1	CommuniGate Pro single-server test, SIP proxying only	1 CommuniGate Pro Server performing all functions.	No	No	No	781.6
R2	CommuniGate Pro Cluster with one Frontend and one Backend, SIP proxying only	1 Frontend (handles all SIP transactions), 1 Backend (performs registration functions, AOR lookup, and cluster management.)	No	No	No	712.9
R3	CommuniGate Pro Cluster with multiple Frontends and one Backend, SIP proxying only	15 SIP Farm Frontends and 1 Backend.	Yes	No	No	6568.4
R4	CommuniGate Pro single-server, SIP calling with registered accounts	1 CommuniGate Pro Server performing all functions.	No	No	Yes	175.8
R5	CommuniGate Pro Cluster with one Frontend and one Backend, SIP calling with registered accounts	1 Frontend, 1 Backend.	No	No	Yes	Not tested (planned but not completed due to time)
R6	CommuniGate Pro Cluster with multiple Frontend and one Backend, SIP calling with registered accounts	15 SIP Farm Frontends and 1 Backend.	Yes	No	Yes	394.0
R7	Full SIP Farm Dynamic Cluster, multiple Frontends and Backends	13 SIP Farm Frontends, 7 Backends.	Yes	Yes	Yes	1049.6

4.2 Run Profile of Test R3

Test R3 is notable for its very high performance rates. As described in the above table, this particular test measures “SIP proxy” performance, and is not limited by the significantly higher I/O required for SIP REGISTER authentication and inbound call lookup by AOR (e.g., <sip:test1@example.lan>) in order to relay to the registered contact address (for example, Contact:<sip:test1@192.168.1.111:5060>).

4.2.1 Run summary

During this run, 17 Linux machines were involved:

- srv1 – srv15 acting as Frontends
- srv20 acting as Backend
- srv21 acting as Load Balancer

6 sipp UAS processes were launched on sipp1, and 6 sipp UAC processes were launched on sipp10 machines.

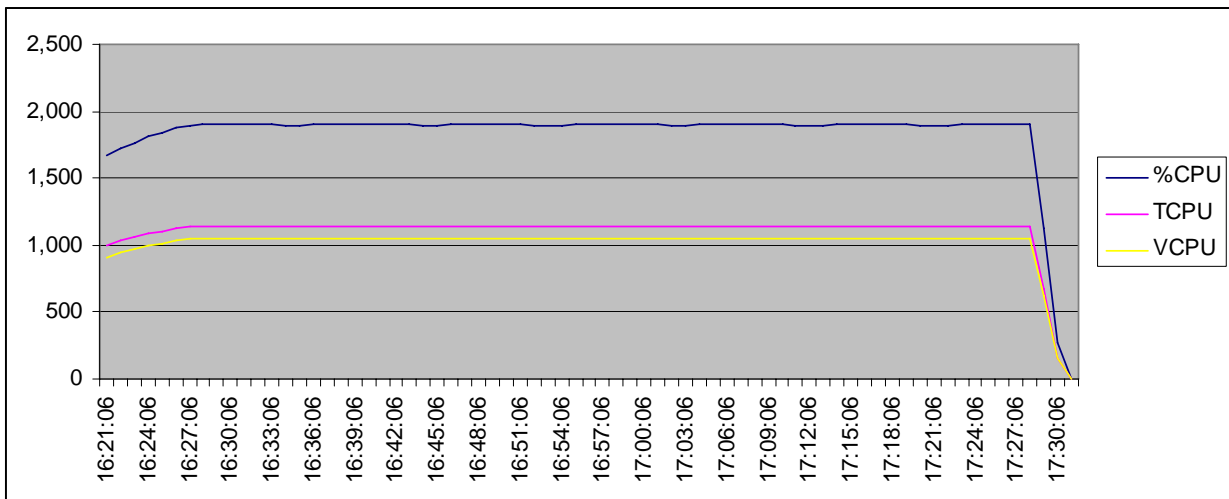
The call rate over 1 hour achieved 6,568.4 new SIP calls/second throughput, with 387,536 concurrent active calls in average.

4.2.2 Run summary table

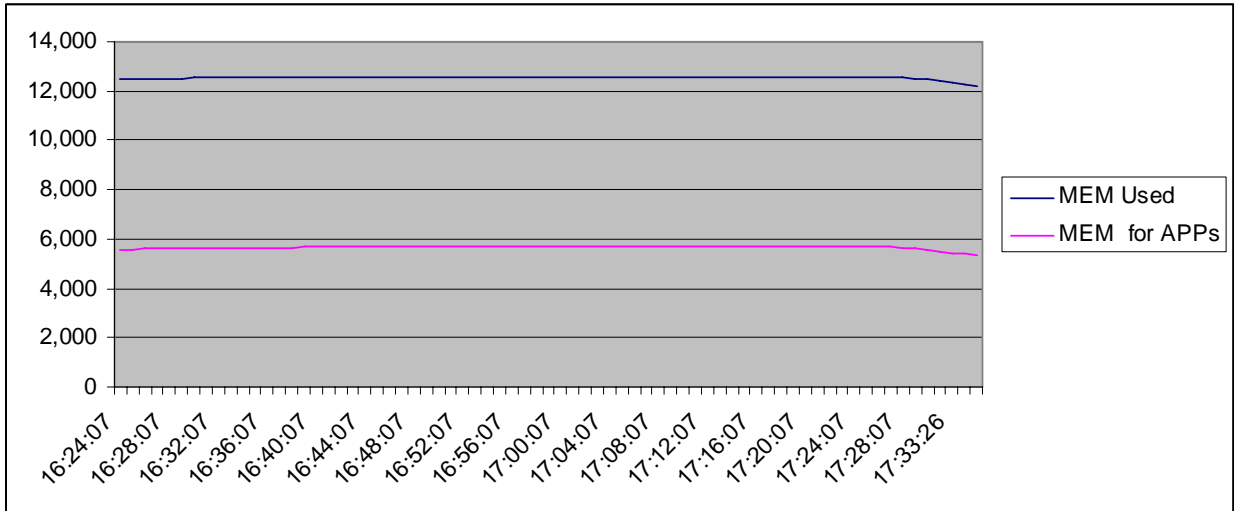
Test Title	R3 (R3210801)
Total Time of Test (mm:ss)	64:50
z/VM Linux machines used for SIPp	10
SIPp UAC processes	35
SIPp UAS processes	35
Total Calls	32,816,231
Calls per second	6568.4
Failed Calls	0
% of Retransmissions (of all packets)	0.108%
SIPp UAC Average Response Time	0.006 seconds (6 ms)
Total CPU Usage in LPAR1	1930.47% (19.3047 CPU out of 24)

4.2.3 Resource consumption

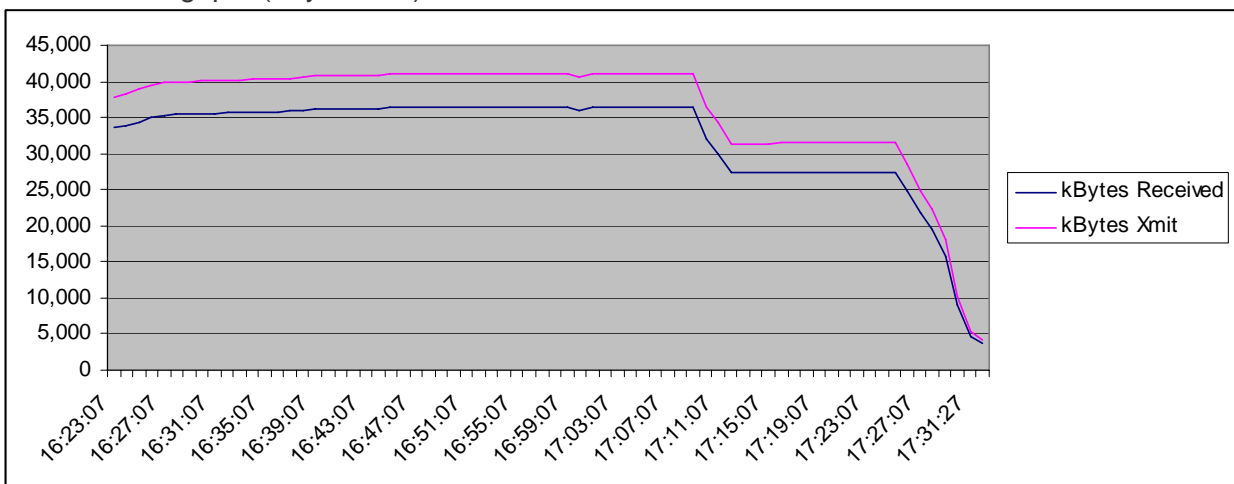
The average CPU utilization was 1930.47%.



Memory consumption (MBs):



Network throughput (kBytes/sec):



5 Conclusions

The following conclusions can be drawn from this performance test, as they relate to the objectives set for the test

5.1 System z exploitation

The System z solution architecture was defined, tested and verified both from functional and performance point of view. Several unique System z related technologies were exploited.

5.1.1 z/VM virtualization

Several advantages were related to the fact that we were running our Linux system under z/VM hypervisor:

- Resource sharing: all of the resources (CPs, memory, disks, networking) were shared between virtual machines
- Idle Linux systems: running idle Linux systems had no impact on the overall performance of the system. This could be used in the real production environment where not all the servers are busy on the same time
- Virtual networking
- Performance Monitoring: Using z/VM Performance Toolkit we were able to know exactly which resources were used by particular virtual machine. In addition the APPLDATA monitoring inside Linux machines was integrated with Performance Toolkit providing us with the most detailed Linux related performance data

5.1.2 Virtual SWITCH

Network throughput is one of the most important factors in the VoIP solutions. Using the Virtual SWITCH feature of z/VM, we were able to:

- share the single physical 10 Gb Ethernet Adapter between all our Linux machines running CGP
- share the other physical 10 Gb Ethernet Adapter between all the Linux virtual servers running SIPp tool
- establish a very fast communication between Front-end and Back-end servers which were using the virtual network

5.2 Scalability

Our tests were focused on the CommuniGate Pro Server application scalability. We have proved that the application was extremely scalable, demonstrating near linear growth in performance as additional SIP Farm Frontend Servers were added to the cluster. When more workload was put on the system (simulating more users and calls), a rapid increase in CPU load and CPU "I/O wait" on the dedicated NAS server, srv20 occurred.

Our tests were focused mainly on the CP consumption. To avoid any memory constraints, our LPARs were configured with enough storage (memory in Linux terms) to avoid significant paging or swapping. Nevertheless, the measurements proved that the memory consumption was not significantly increasing during the different runs.

5.3 Stability

The stability of the system was also verified. Once the system was running under stress conditions, all the virtual servers were handling end-users activity in very constant way (and still recording the performance data). We never had any server critical failures, such as crashes, hangs, or data space corruption.

5.4 Lessons Learned

As always, there are items which could be improved upon, given another chance to perform this or a similar performance test.

- a) Overall performance could have been improved with better "access" to the storage, and it is quite likely that the performance improvements in doing so would be significant, improving overall performance 100% or more. The underlying DS8100 had 116 disks allocated to the CommuniGate environment and never went much above 5800 IOPS, which was only about 30% capacity of those 116 disks. The NAS setup (using just a Linux z/VM instance instead of a true NAS server with optimized cache) was a clear bottleneck. An Enterprise-quality NAS head to the array would likely have provided a significant performance improvement. From CommuniGate Systems' experience, using Linux as an NFS server is rarely if ever recommended, as it cannot match the performance and concurrency capabilities of a dedicated NAS system.
- b) Problems encountered with OCFS2 as a Cluster File System for use in a CommuniGate Pro Dynamic Cluster forced the testing to use the NAS solution described above. A high-performance Cluster File System with native access to the DS8100 SAN would likely have increased disk I/O "access" performance significantly. A Cluster File System on the storage system would likely have resulted in improvements as well..

5.4.1 Testing notes

- a) It is worth noting that shortly after the performance testing described in this report was completed, IBM submitted a large number of code changes to the sipp project. These changes were submitted due to performance and efficiency problems found in sipp under high loads (exactly the type of loads demonstrated in this performance test,) and these identified performance issues were consistent with apparent limitations of sipp encountered in this performance test. Using a more recent version of sipp would likely have improved vertical scaling of the sipp infrastructure by allowing more "Calls Per Second" from each sipp instance, rather than having to scale horizontally using many sipp instances none of which can communicate with the others or be managed as a group. Though not yet confirmed, it appears that using a more recent version of sipp (post-IBM code submittal) would greatly improve overall measured results in this test, even if performed identically to the tests documented here.
- b) Please contact IBM or CommuniGate Systems if there is interest in the SIPp configuration files and start scripts used in these tests.

5.4.2 General recommendations

The z9 machine used for the tests was equipped with Central processors (CP) only. In a real environment we recommend to use the Integrated Facility for Linux (IFL) processors for z/VM & Linux partitions.