PROMPT ENGINEERING

liquid thoughts. solid experience

Performance/Cost Estimation of a Virtual Cluster Running HyperWorks

Roberto Vadori, Vincenzo Stirparo

PROMPT ENGINEERING

liquid thoughts. solid experience

Adriano Ajmar, Claudio Ambroggio, Gianluca Zaninelli

Reply

Living network

Contents

- A short description of the Amazon installation
- Networking and communication channels between Amazon and Reply
- Client-server framework and licensing
- Cluster deployment
- Cluster Test: performances, remote visualization, file transfer
- Costs estimation

Amazon Installation (I)

Amazon Web Services offers an “Infrastructure as a Service” **(IaaS)**, that is to say **remote hardware resources**, driven by an Operative System chosen (from a quite wide range) by the user.

Virtual hardware used for the suite of tests is an **Intel Xeon X5500,** 64 bit, dual- and quad-cores, 2.67 GHz clock, 17 GBytes of RAM, 100 MBytes Ethernet interface, but ***“Of course, the actual hardware your server runs on probably looks nothing like this; it’s just a way of describing your allocation”,*** *Nemeth et alii,* Unix and Linux System Administration Handbook, 4th Ed., 2010, pg. 1008

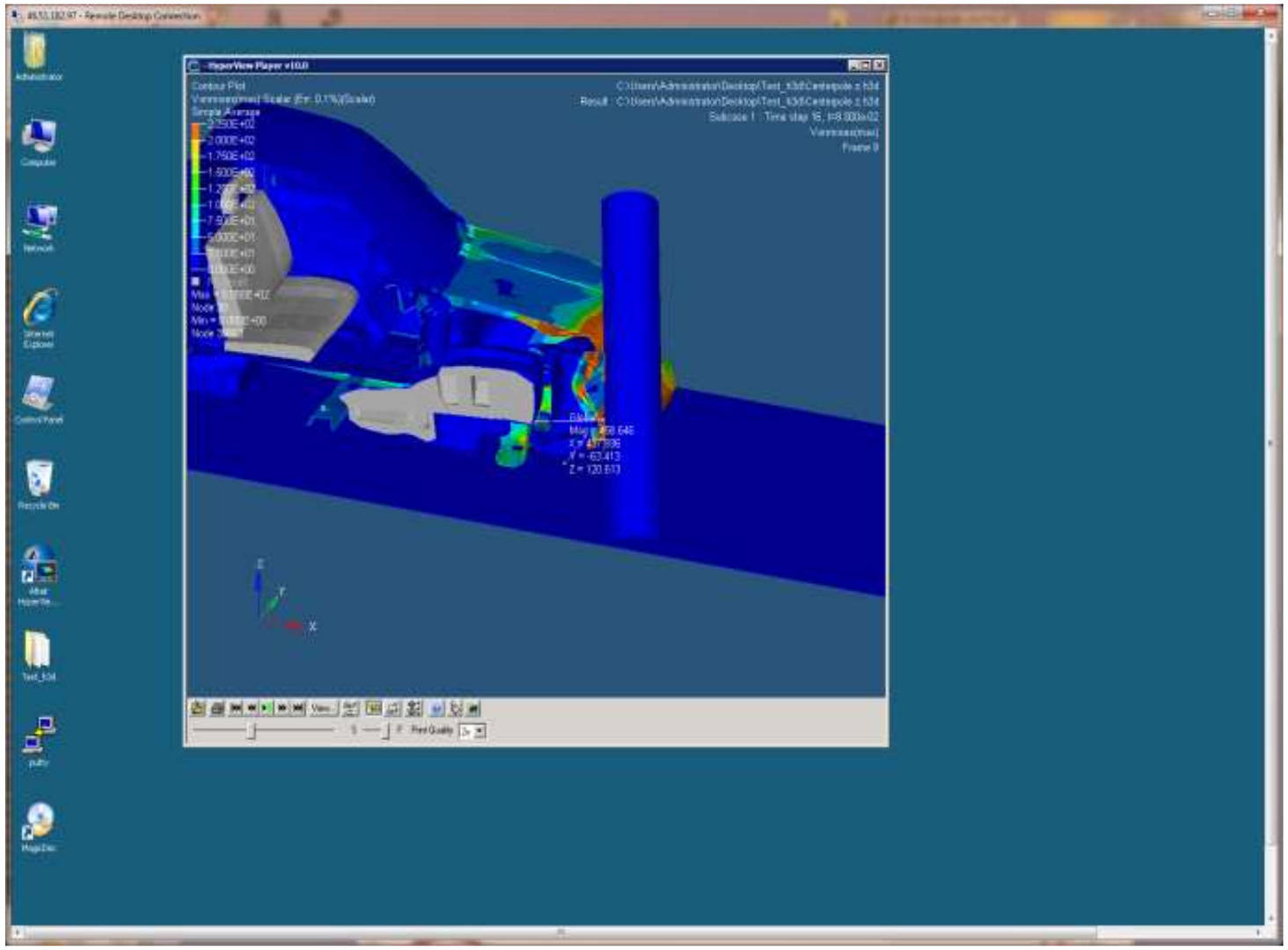
Operative system of choice is Linux (Fedora 8 distribution), no graphical capabilities and no X, command-line interface only. Just a **“number crunching”** setup.

Installed Software is the suite of Altair's solvers (Optistruct and RADIOSS) along with the queue management system PBSpro; the transaction between license server and clients is performed by a server plugged into the Reply intranet.

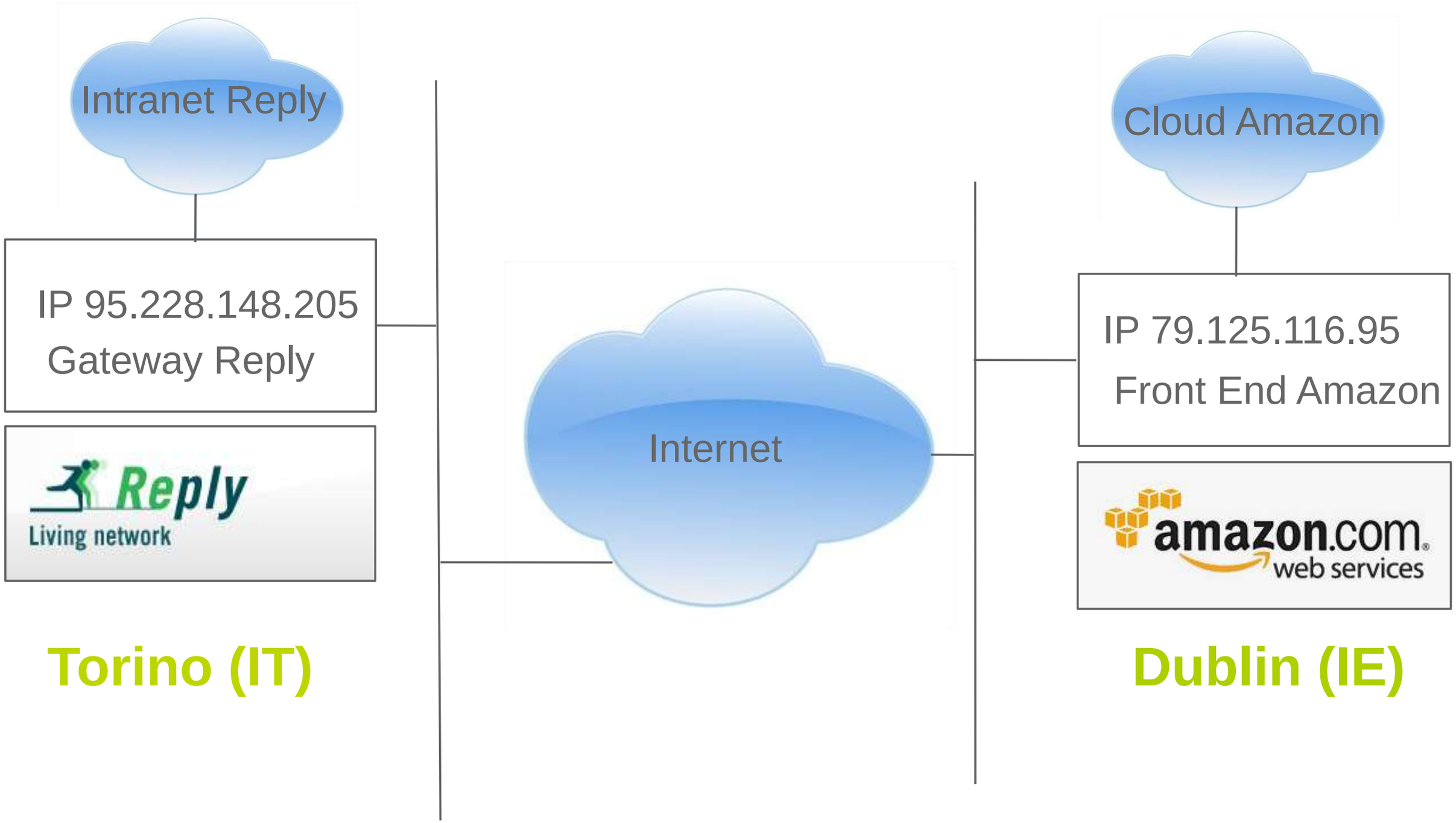
Amazon Installation (II)

Graphical capabilities are managed by another virtual machine, always on the Amazon side: a Windows Server with the same hardware nominal characteristics of the computing nodes. In this case chosen Operative system is Windows (Windows Server 2008)

Installed Software is the suite of Altair's post-processors (HyperView and HyperView Player). Even in this the transaction between license server and clients is performed by a server plugged into the Reply intranet.



Network Infrastructure and Links (I)



Torino (IT)

Dublin (IE)

Reply Network 95.228.148.0

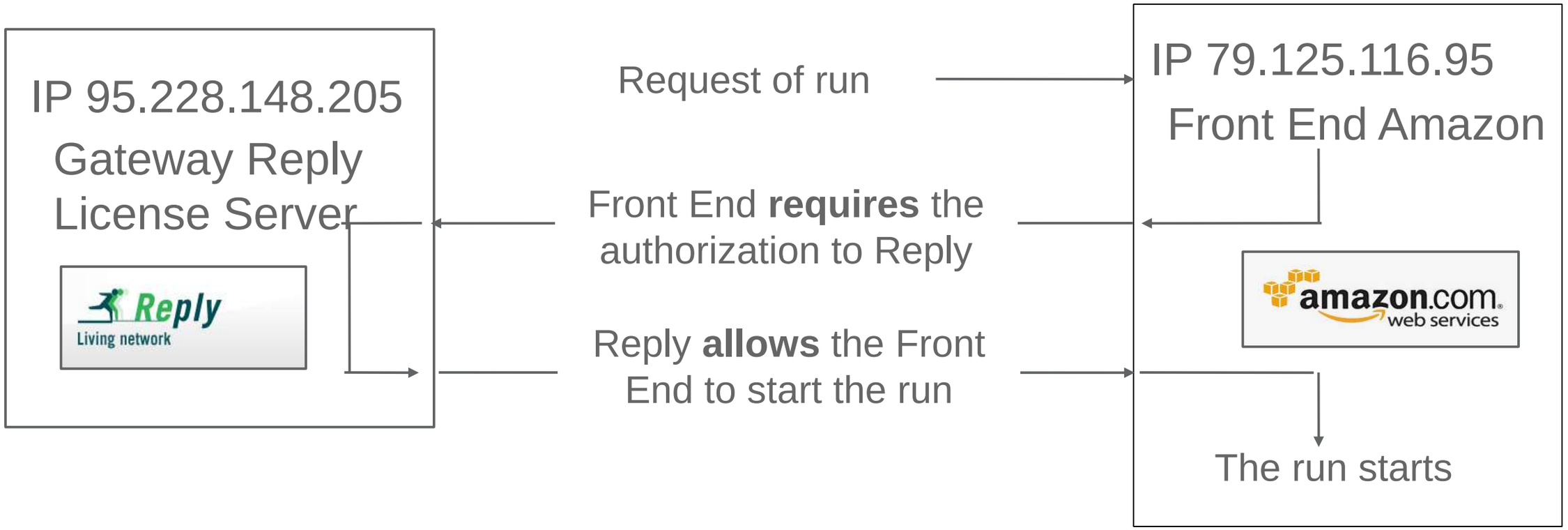
Amazon Network 79.125.116.0

Client-Server License Framework

Simply stated:

LICENSE SERVER MUST ALWAYS REMAIN UNDER THE EXCLUSIVE CONTROL OF REPLY

Beside of security reasons a technical issue arise, too: at every reboot of the Amazon-side machines, their MAC address change, posing a (useless) challenge in the remote administration of the license file. So we won't and we can't keep the license server in the Amazon network.



Cluster Deployment

The cluster deployment is fully driven by a set of scripts, each one of them...

- Performs the **installation of a “Front End”**
- Performs the **installation of a set of remote nodes**
  - Reboots the remote nodes. **At the reboot**, each node, by means of a client-server simple applications (see e.g. Stevens, UNIX Network Programming, Volume 1, Second Edition: Networking APIs: Sockets and XTI, Prentice Hall, 1998) **communicates its own IP address to the Front End**
- Build and deploy a common “hosts” file to assign a standardized name to each node
- Creates an “**user environment**” on the Front Enad and each node...
- Mounts all the shared and common directories to each of the nodes...
- **Starts PBSpro** framework (server on the front-end, MOM on each node)

The whole process, for a cluster of 10 quad-cores nodes, takes **less than 24 minutes**. That is to say,

AFTER 24 MINUTES YOU HAVE A FULLY OPERATIVE “40 SLOTS” CLUSTER

Virtual Remote System

IP 79.125.116.95
Front End Amazon
Server PBS
Server NFS

Queue mgr: PBS
MPI interf: HP-MPI

IP 10.228.125.x
Node_01
Client PBS
Client NFS

IP 10.228.125.x
Node_02

IP 10.228.125.x
Node_0..

IP 10.228.125.x
Node_0N

amazon.com web services

Dublin (IE)

Rete 79.125.116.0

IP 10.228.125.x
Windows Server
Graphical Machine

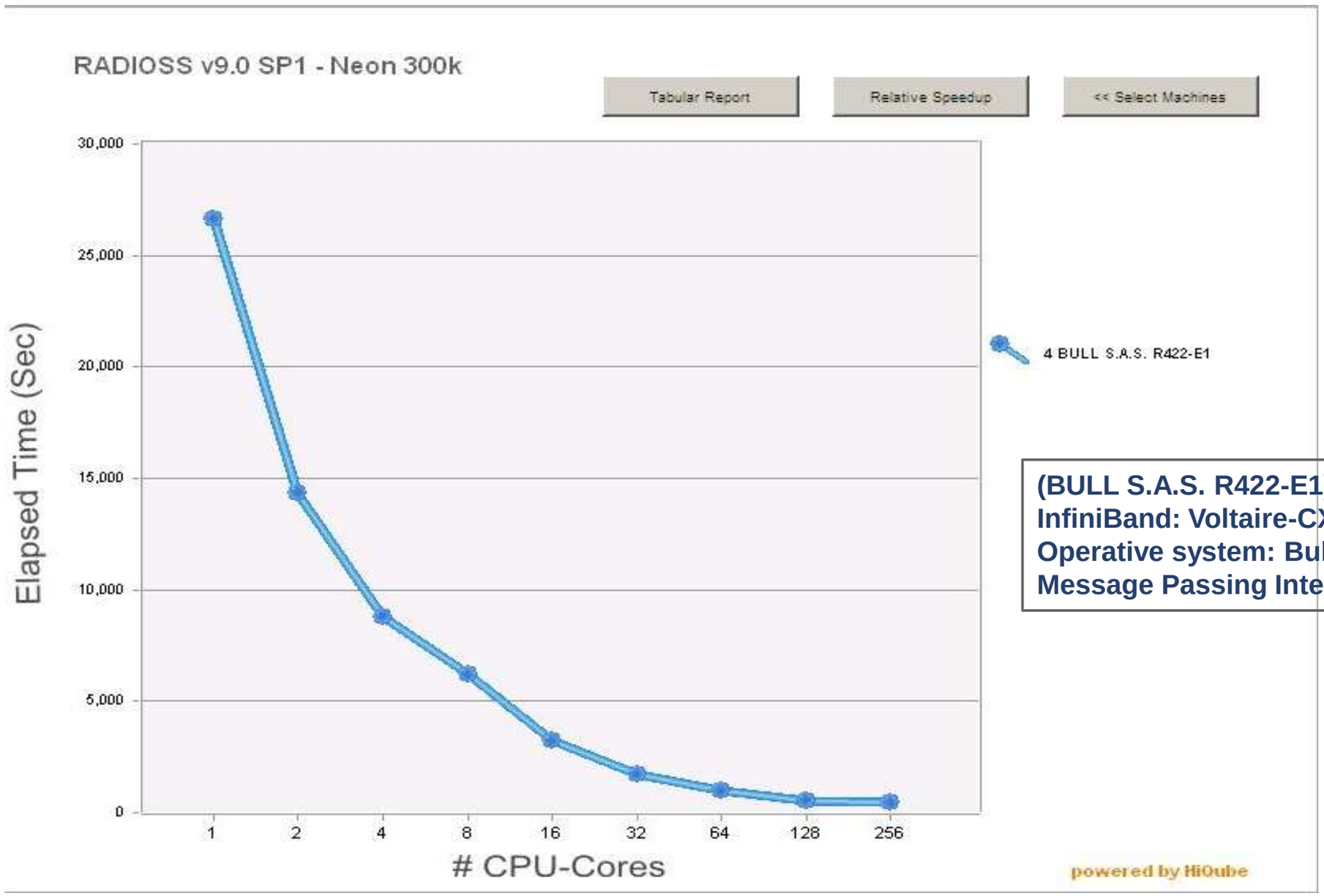
Rete 10.228.125.0

Installed Software

On the Front End we installed **RADIOSS, Optistruct and MotionSolve**.
Queues are managed by **PBS-Pro** while the graphical visualization is performed by a Windows Server virtual Machine running **HyperView.**

RADIOSS runs (I)

Altair's official benchmark: Neon_300k", an explicit input deck of a full crash; 300.000 shell elements, about 500.000 dof. Comparison baseline was chosen having in mind a similar architecture:

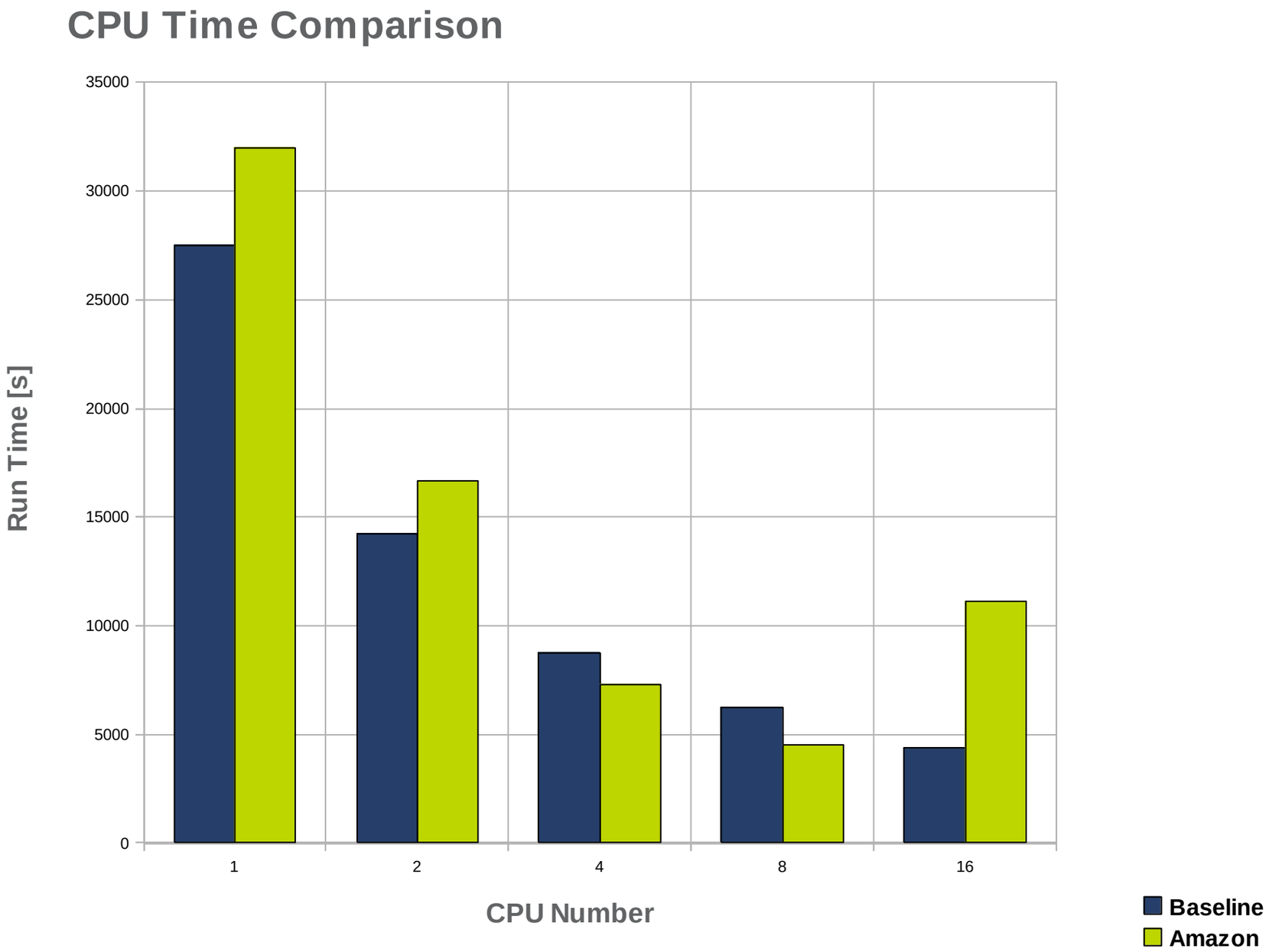


(BULL S.A.S. R422-E1) Intel Xeon E5462@ 2.8GHz - InfiniBand: Voltaire-CX-DDR-4x-OFED1.3 - Operative system: Bull AS 5v1.1 - Message Passing Interface: HP-MPI_2.2.7

RADIOSS runs (II)

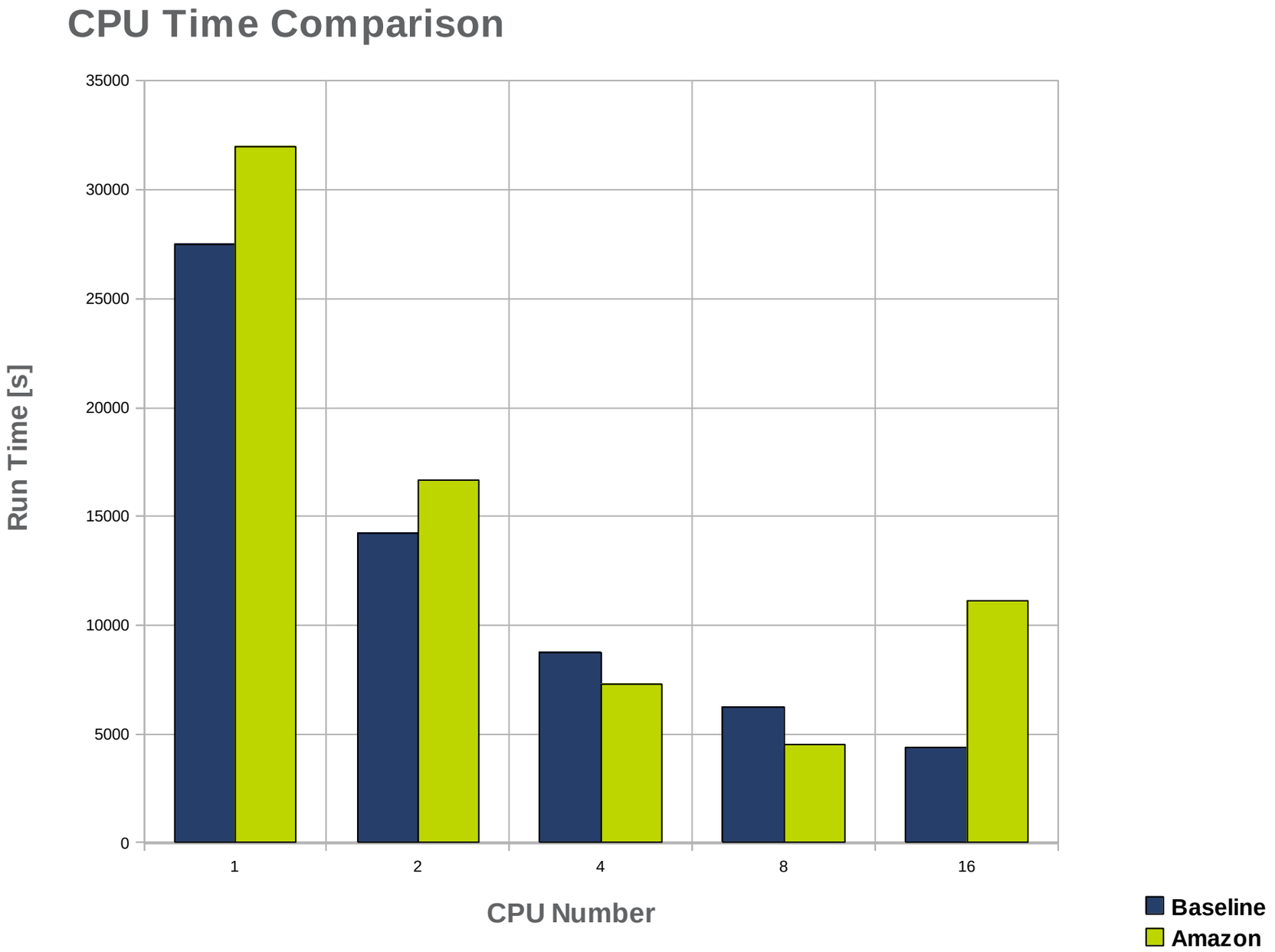
Making the comparison as fair as possible, we scaled the execution times by the ratio of the clocks :

- F = Clock_BULL/Clock_Amazon = 2.89/2.67 = 1,08



RADIOSS runs (III)

Amazon starts not with a high pace, but as the CPU number increases behaves better and better till 8 CPUs. A remarkable drop down of performances is obtained with sixteen virtual CPUs. **There is a reason!** But an Amazon based cluster scales quite well



RADIOSS runs (IV)

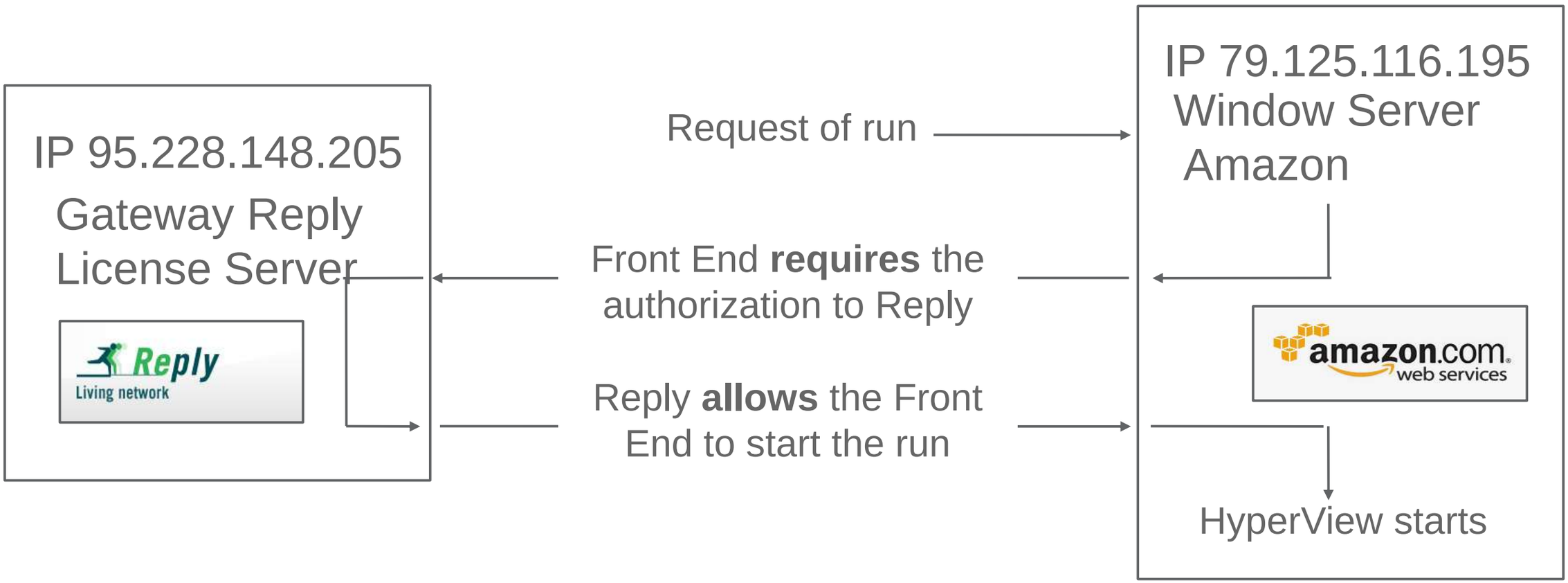
The transfer rate between Reply to Amazon (upload of input decks) is reasonably fast, given that we have to transfer just only few text files: using SCP with no download manager, upload is less responsive than downolad, but still reasonable.
Download of the resuts is a bottleneck indeed: other solutions are to be found.

```
Transferred: 22394880 KBytes
Time: 3'13" = 193 s
Average: 21870*1024/193 = 113.30 KB/s
```

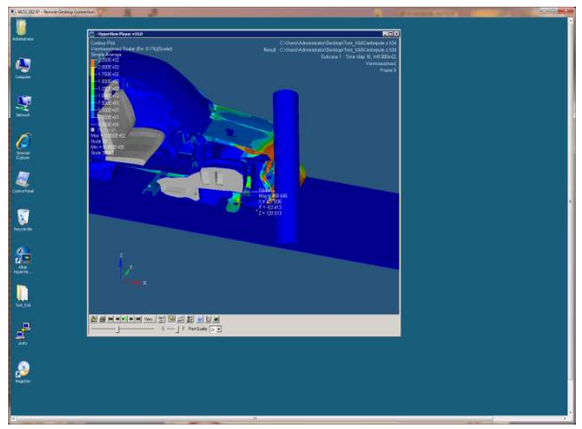

```
Max: 513 KB/s
Min: 170 KB/s

Transferred: 145858560*2 = 291717120 Bytes
Time: 17'23" = 1043 s
Average: 273.35 KB/s (1K = 1024 B)
```

RADIOSS post processing (I)

The proposed solution is to visualize the results on a remote Amazon virtual machine running Hyperview:



Users



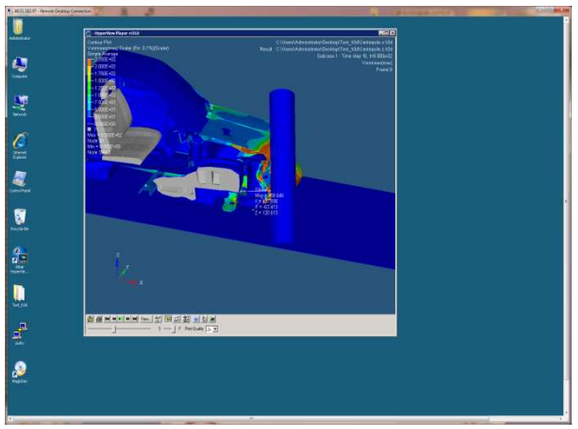
RADIOSS post processing (II)

When the results are considered satisfactory, binary result files are downloaded in the customer data storage with a FTP server running on the virtual server.

Front End → View/Ftp Server: 277 MB in 130 seconds via LAN winscp

View/Ftp Server → Local Machine: 277 MB in 665 seconds via ExpeDat (Data Expedition, inc) ftp server and mobile connection: HDSPA (avg 3,49 Mb/s)

View/Ftp Server → Local Machine: 277 MB in 360 seconds via ExpeDat (Data Expedition, inc) ftp server and ADSL connection, bandwidth 7Mb (avg 6,45 Mb/s)



Costs Estimation (I)

As an example we imagine a set of “instances” as of

```
-----------------------------------------------------------
High-Memory Extra Large Instance 17.1 GB memory,
6.5 ECU (2 virtual cores with 3.25 EC2 Compute Units each),
420 GB of local instance storage, 64-bit platform
-----------------------------------------------------------
```

This instance has a hourly cost of $0,57/h for the installed configuration.

To this hourly cost we have to take into account the upload, storage ($0,11 GB/month) and download of the various files.

WE DEFINE A “RUN”

- **THE UPLOAD OF AN INPUT DECK,**
- **THE EXECUTION,**
- **AND THE DOWNLOAD OF THE RESULTS**

In other words, we evaluate the costs of a run not taking into account licensing costs (too many variables).

Costs Estimation (II)

INPUT STORAGE	Input [B]	Input [GB]	Cost [$]
Optistruct	31369249	0,03	0,0032
Radioss_1CPU	30621682	0,03	0,0031
Radioss_2CPU	30621682	0,03	0,0031

INPUT TRANSF	Input Rate [KB/s]	Upload Time [s]	Upload Time [min]	Upload Time [h]	Cost [$]
Optistruct	113,3	270,38	4,51	0,08	0,04
Radioss_1CPU	113,3	263,94	4,40	0,07	0,04
Radioss_2CPU	113,3	263,94	4,40	0,07	0,04

RUN TIME	Time [s]	Time [h]	Cost [$]
Optistruct	5465,00	1,52	0,87
Radioss_1CPU	31982,70	8,88	5,06
Radioss_2CPU	16668,88	4,63	2,64

	Total Time [s]	Total Time [h]	Total Cost	Hourly Cost [$]	Hourly Cost [€]
Optistruct	11709,1	3,25	2,03	0,62	0,45
Radioss_1CPU	37372,91	10,38	6,07	0,58	0,42
Radioss_2CPU	22059,09	6,13	3,64	0,59	0,42
Radioss_4CPU	7543,94	2,10	1,45	0,69	0,49
Radioss_8CPU	4813,94	1,34	1,05	0,79	0,56

	Input Time cost	Input Storage Cost	CPU Cost	Output Time Cost	Output Storage Cost	Total Cost
Optistruct	0,04	0,0032	0,87	0,95	0,17	2,03
Radioss_1CPU	0,04	0,0031	5,06	0,81	0,15	6,07
Radioss_2CPU	0,04	0,0031	2,64	0,81	0,15	3,64

	Total Time [s]	Total Time [h]	Total Cost	Hourly Cost
Optistruct	11709,1	3,25	2,03	0,62
Radioss_1CPU	37372,91	10,38	6,07	0,58
Radioss_2CPU	22059,09	6,13	3,64	0,59

Conclusion (I)

An HPC solution based on Amazon turns out to be:

- Scalable
- Reliable
- Easily deployed

Hourly cost is very interesting indeed. It varies from **€ 0.42** to **€ 0.56** per hour. The costs of similar services (in Italy) spans from **€ 2.40** to **€ 3.00** per hour (software inclusive).

A SOLUTION BASED ON AMAZON IS COSTLY EFFECTIVE

Hourly cost does not vary even if with more than one instance, and the excellent scalability allows to reduce extimated time of execution almost in a inversely proportional ratio.

Instance cost is “on demand”, that is to say hourly costs are charged just on only needed, active instances.

HARDWARE INFRASTRUCTURE RELATED COSTS ARE ALMOST ELIMINATED

Prompt Engineering S.r.l.

Operative Site
Environment Park - Via Livorno 60 – 10144 Torino

Contacts
roberto.vadori@prompt-engineering.com
vincenzo.stirparo@prompt-engineering.com

www.prompt-engineering.com