



Installare GNU/Linux su un ampio numero di client con SystemImager e BitTorrent

Andrea Righi <a.righi@cineca.it>

- You have to install a huge number of computers:
 - ✓ PC labs,
 - ✓ Server farms,
 - ✓ HPC clusters,
 - ✓ Complex grid-computing environments,
 - ✓ Etc.
- But you don't have enough time!

Some practical examples



- SystemImager is a software which automates GNU/Linux installs, software distributions and production deployment

- Support all Linux distributions
- Support a large number of architectures
- Make it solve massive installation problems
- Create a centralized point of installation and maintenance

- System Installation
- System Updates
- Build replicants of machines
- File system or block device migration

- It's open-source (GNU GPL v2)
- File-oriented approach:
 - ✓ Distribution agnostic
 - ✓ Hardware independence
 - ✓ Filesystem independence
 - ✓ Plain filesystem dump: exclude swap space or unused partitions
 - ✓ Block device independence
 - ✓ Live customization (manipulate cloned filesystems directly)



Basic concepts

- Image:
 - ✓ Live snapshot of a machine containing files and directories from the root of that machine's filesystem
 - ✓ *chroot*-able filesystem stored in `/var/lib/systemimager/images/$NAME`
 - ✓ Examples:
 - ✓ `/var/lib/systemimager/images/RHEL4`
 - ✓ `/var/lib/systemimager/images/Ubuntu_gutsy`
 - ✓ `/var/lib/systemimager/images/HPC_1.0`
 - ✓ ...

- Image Server:
 - ✓ a server that has all the images available for the installation
 - ✓ “Jukebox” of images



- Client:
 - ✓ a machine to be auto-installed with a (single) pre-selected image
 - ✓ Example: the dancing penguins are the clients :-)

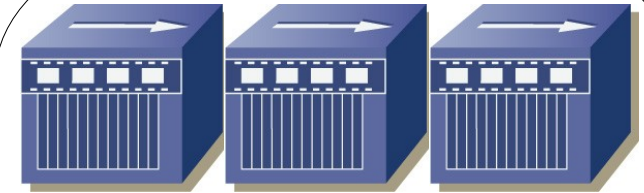


- Transport:
 - ✓ the protocol used to distribute images from the image server to the clients
 - ✓ *push/pull/p2p* approach
 - ✓ Examples:
 - ✓ rsync, multicast, SSL, BitTorrent, ...

**Image Server
(SystemImager)**



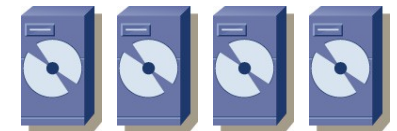
Clients



HPC clusters



HA-clusters

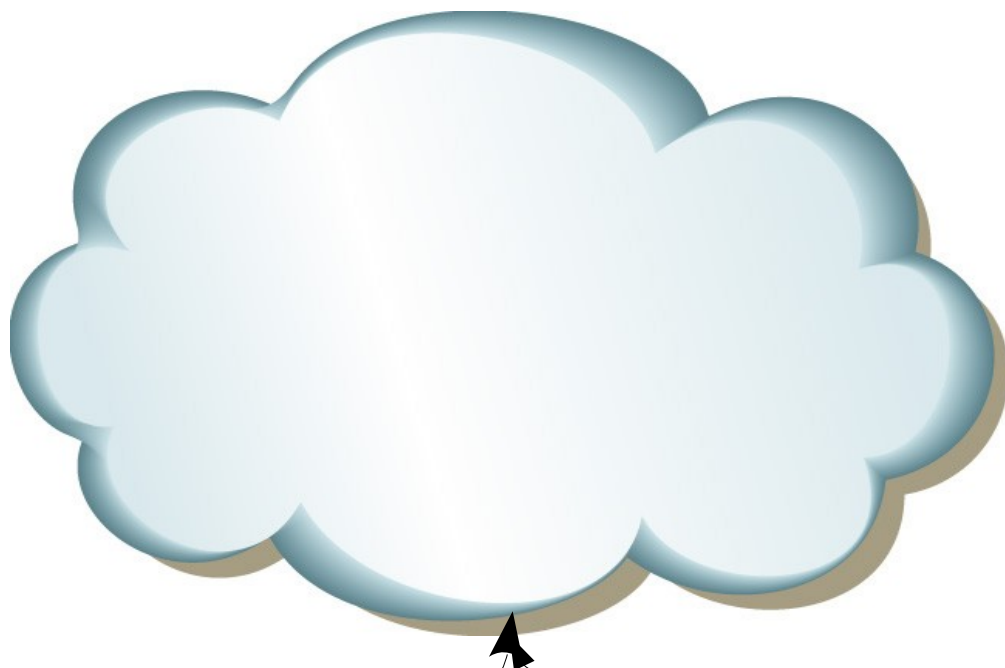


Web farms



PC labs

Image Server
(**SystemImager**)



Golden client



Software
(Debootstrap, YaST,
yum, **SystemInstaller**, ...)

Clients



HPC clusters



HA-clusters

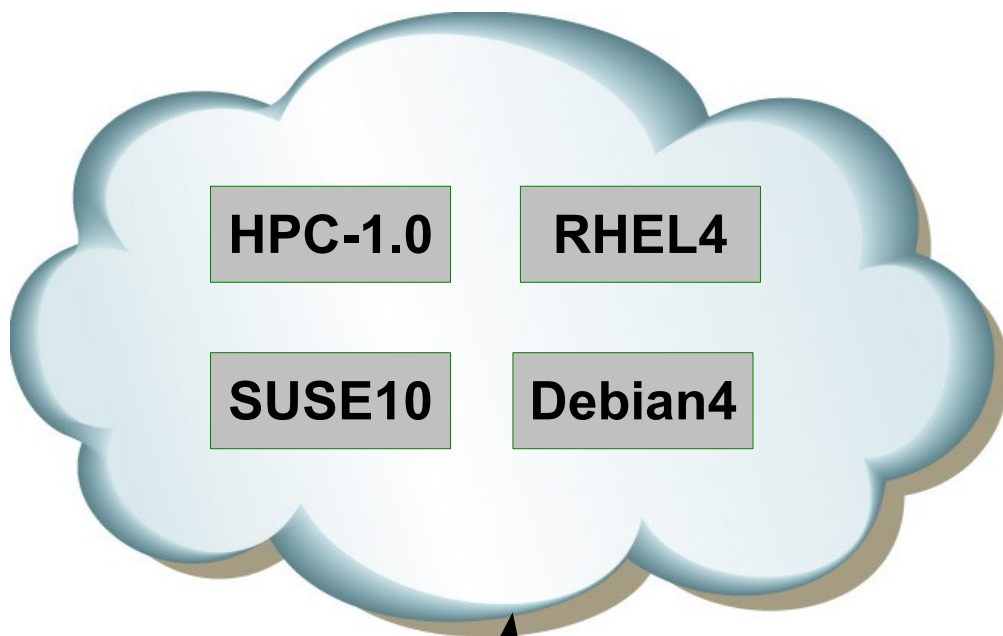


Web farms

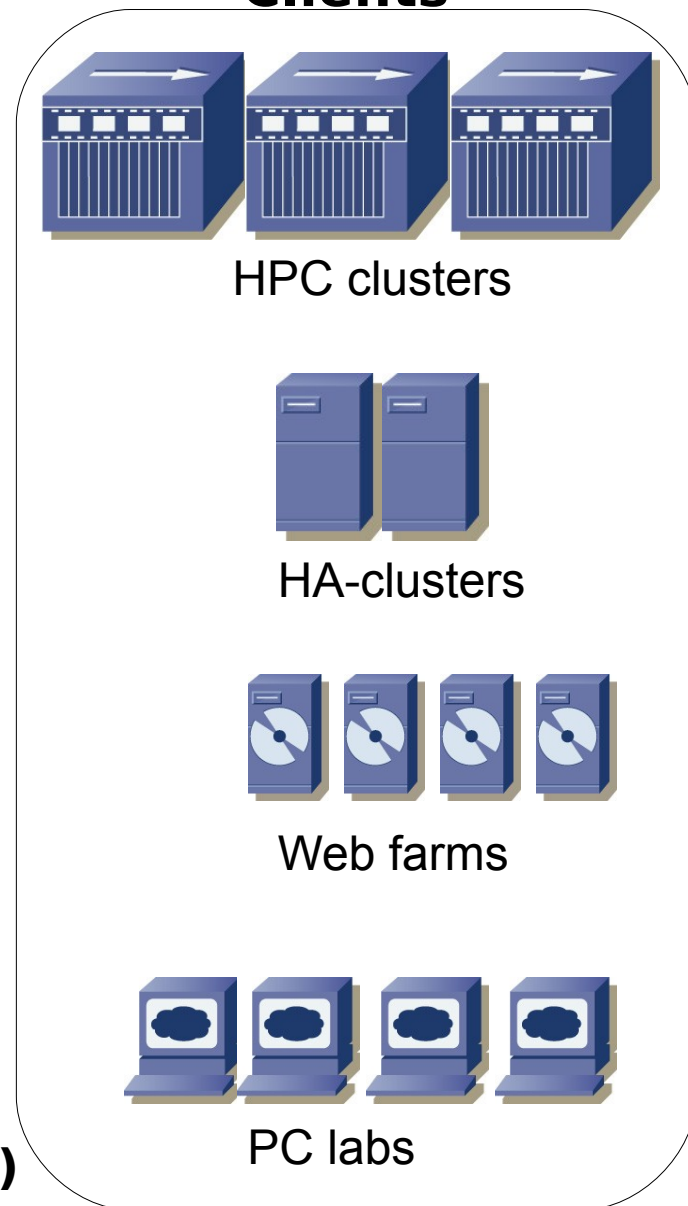


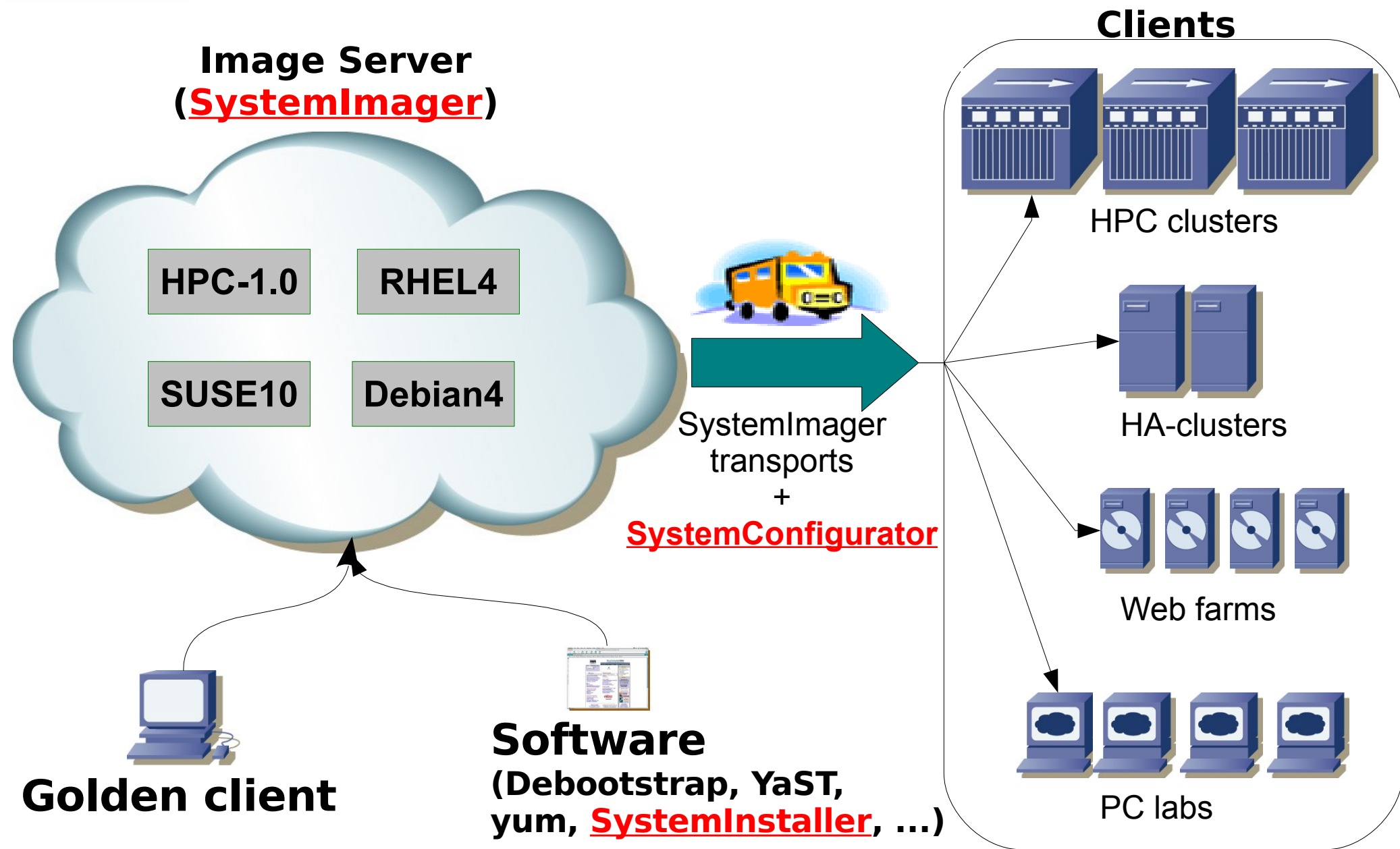
PC labs

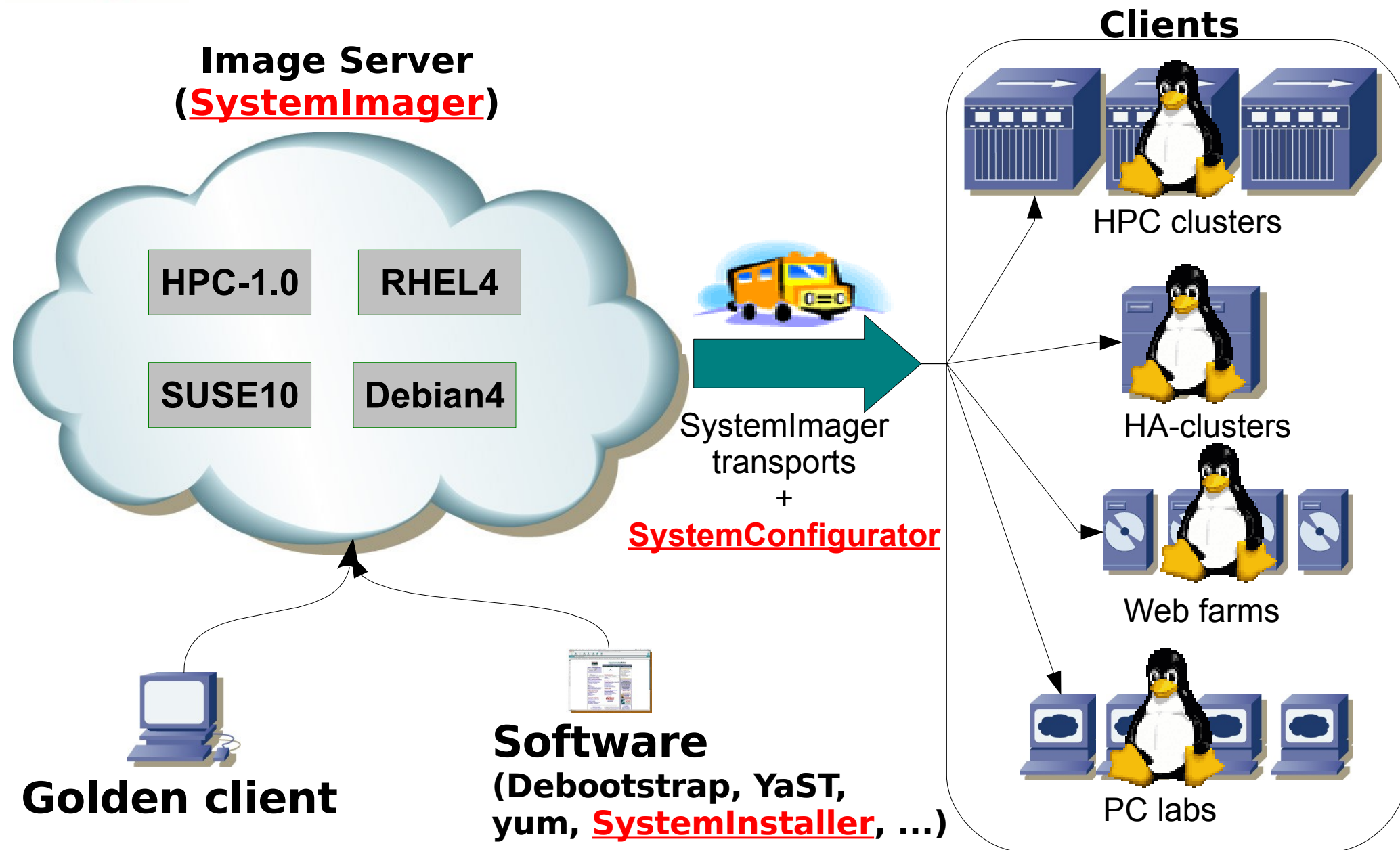
Image Server (SystemImager)



Clients



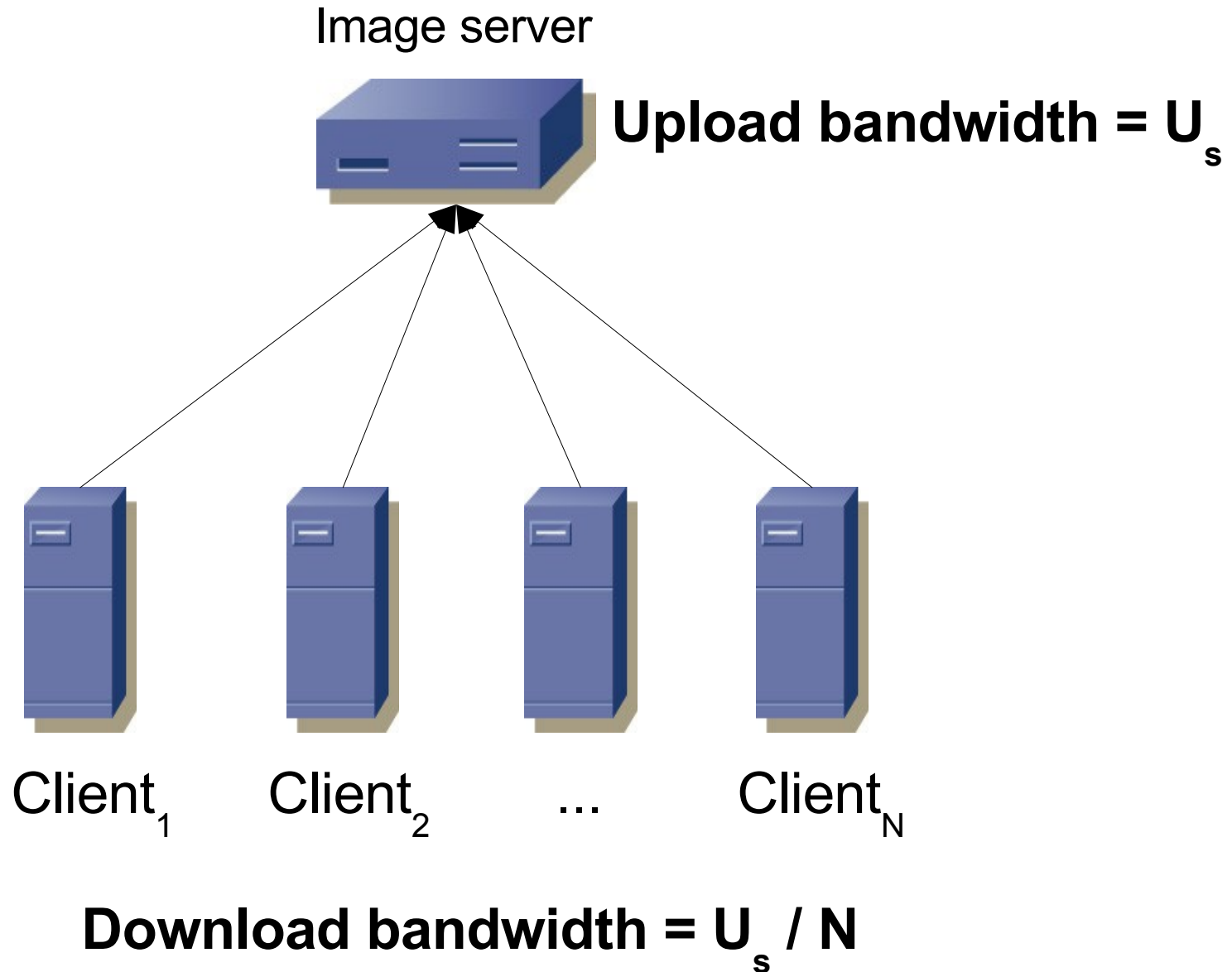






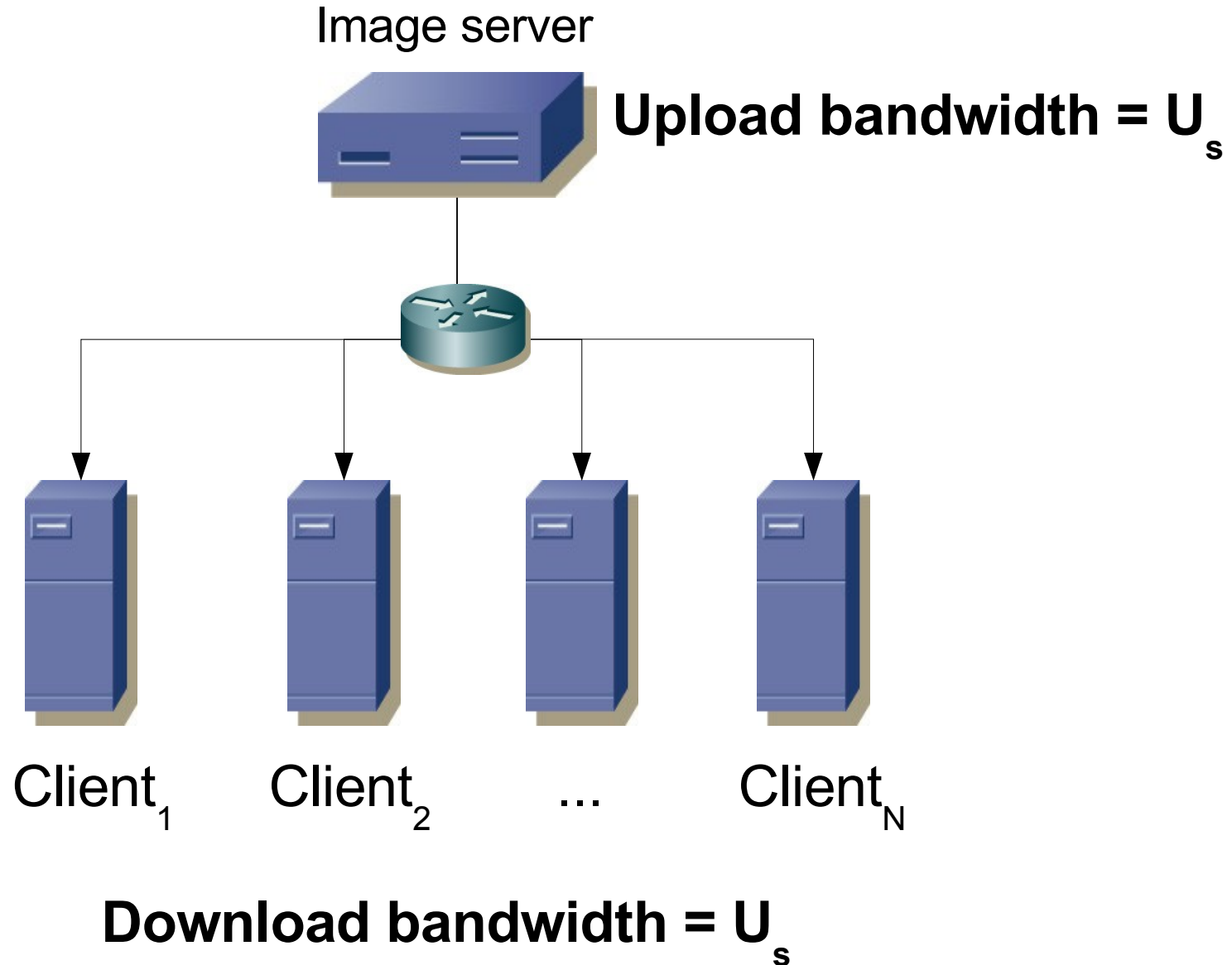
Transports

- rsync (plain / SSL encrypted):
 - ✓ Client-server approach
 - ✓ Limited in scalability
 - ✓ Limited in reliability with a lot of clients
 - ✓ Max Theoretical Bandwidth: Us / N



- Flamethrower:
 - ✓ Multicast approach:
 - ✓ Perfect scalability
 - ✓ But limited in reliability
 - ✓ Max Theoretical Bandwidth: *Us*

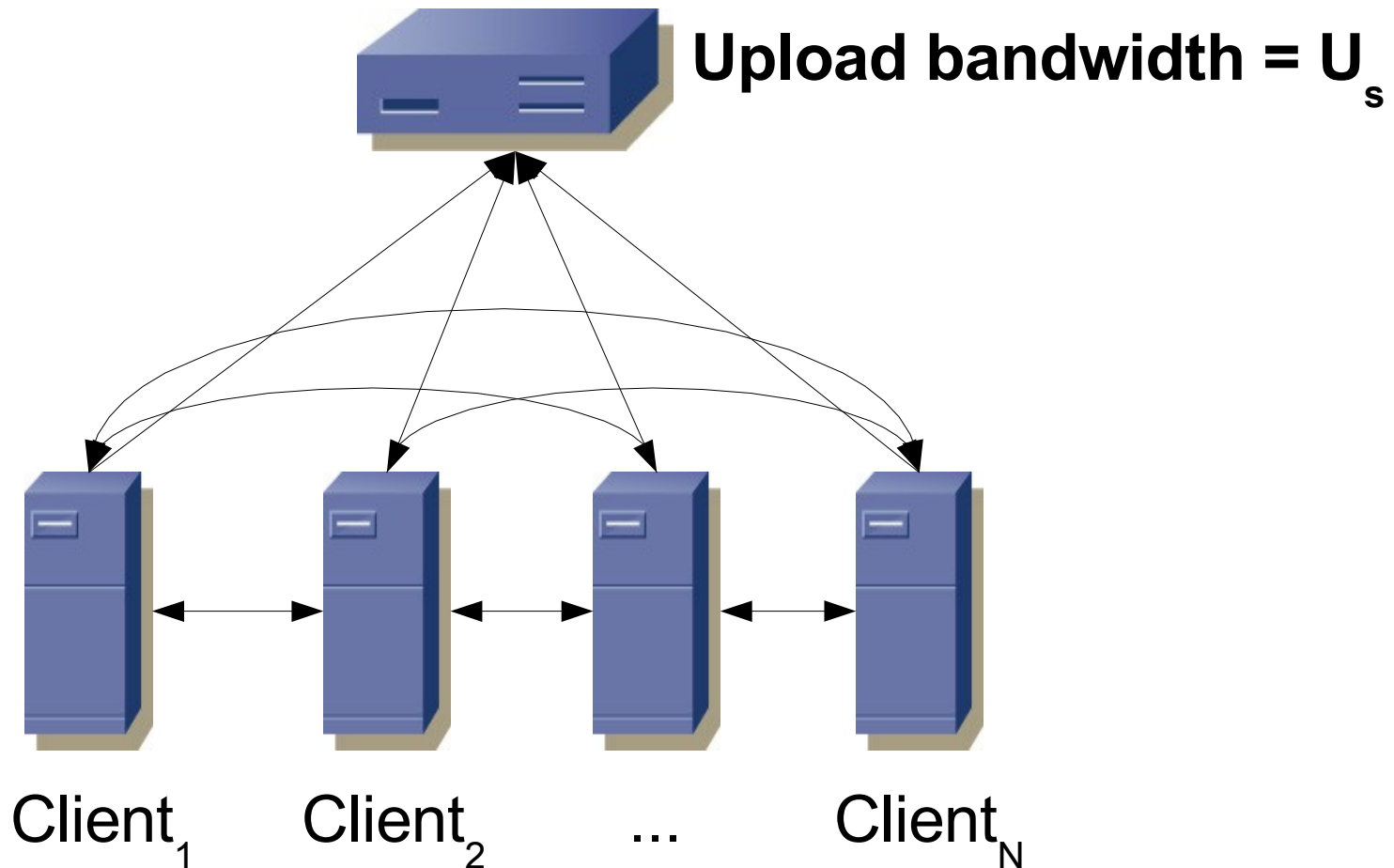
Multicast (Flamethrower) diagram



- BitTorrent is a TCP/IP p2p oriented protocol designed for transferring files
- Peers connect to each other directly to send and receive chunks of data
- There is a central server (tracker) which coordinates the action of all such peers
- The tracker does not have any knowledge of the contents of the files being distributed
- Users upload (*transmit outbound*) at the same time they are downloading (*receiving inbound*)

- BitTorrent:
 - ✓ p2p approach: scalability && reliability
 - ✓ Qiu and Srikant model
 - ✓ Total upload rate: $\mu(\eta x(t) + y(t))$
 - ✓ *Steady state:*
 - ✓ $x(t)$ downloaders $\Rightarrow \frac{dx(t)}{dt} = 0$
 - ✓ $y(t)$ uploaders $\Rightarrow \frac{dy(t)}{dt} = 0$
 - ✓ Max Theoretical Bandwidth:
 - ✓ U_s (steady state)

Image server (+tracker)



Download bandwidth = U_s (in steady state)

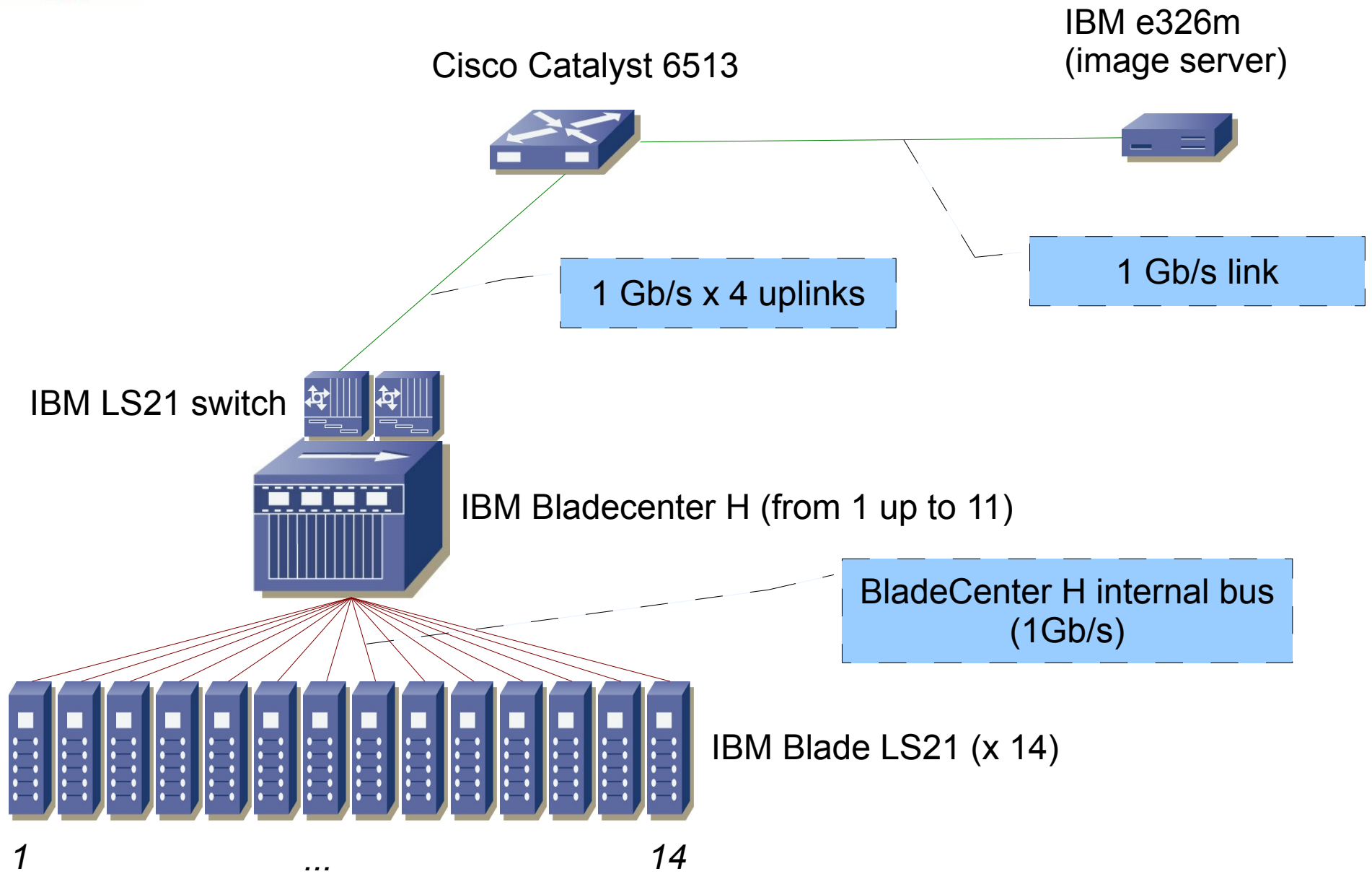


Testbed environment



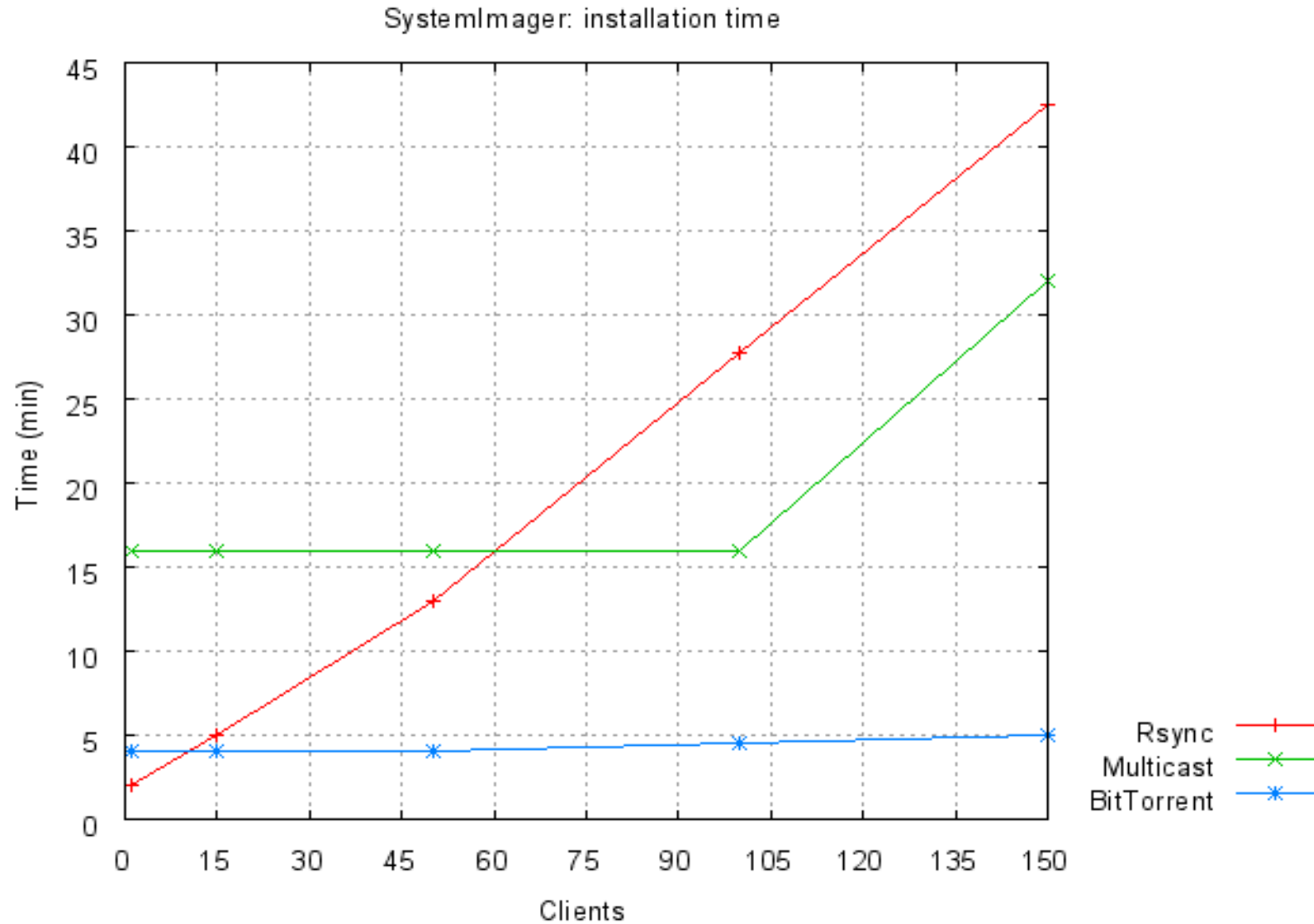
- IBM BCX/5120, with 5120 cores, is the largest computer in Italy for Scientific Computing
- 2 dual-core AMD Opteron(tm) 2.4GHz, 8GB RAM per node
- It is the 44th most powerful computer in the world (TOP500).

BCX network topology

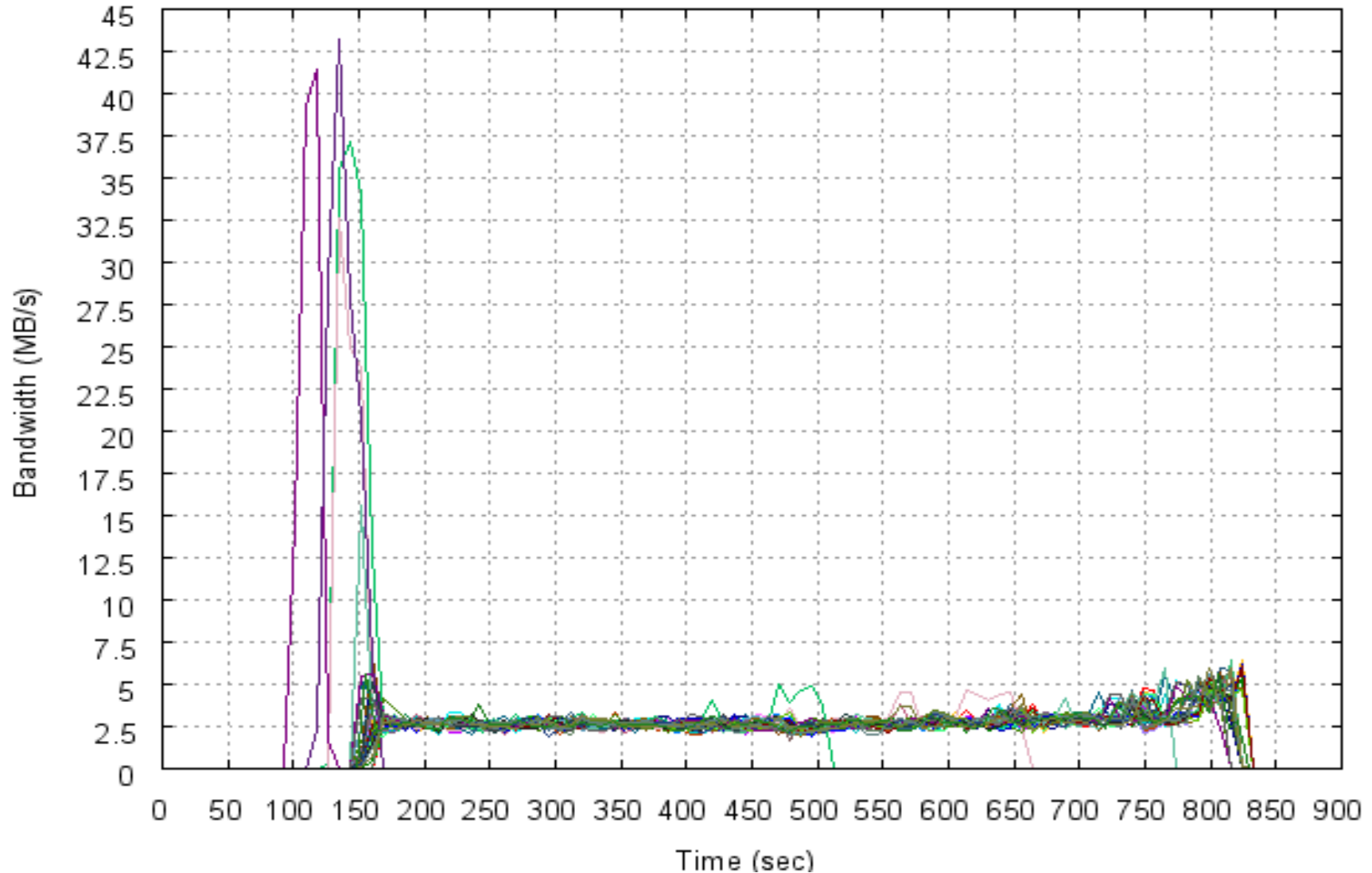




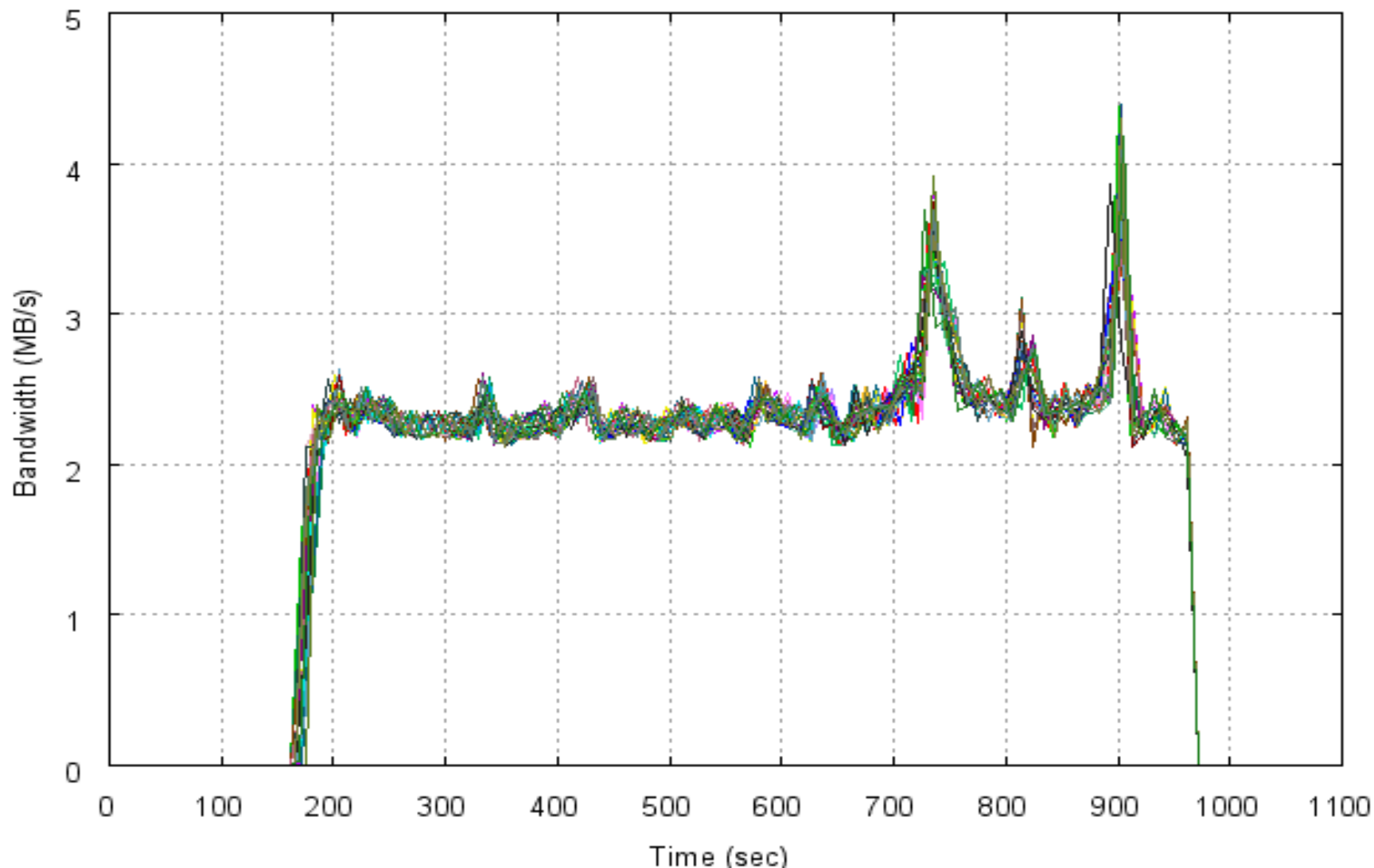
Experimental results



SystemImager (rsync transport): installation of 50 clients

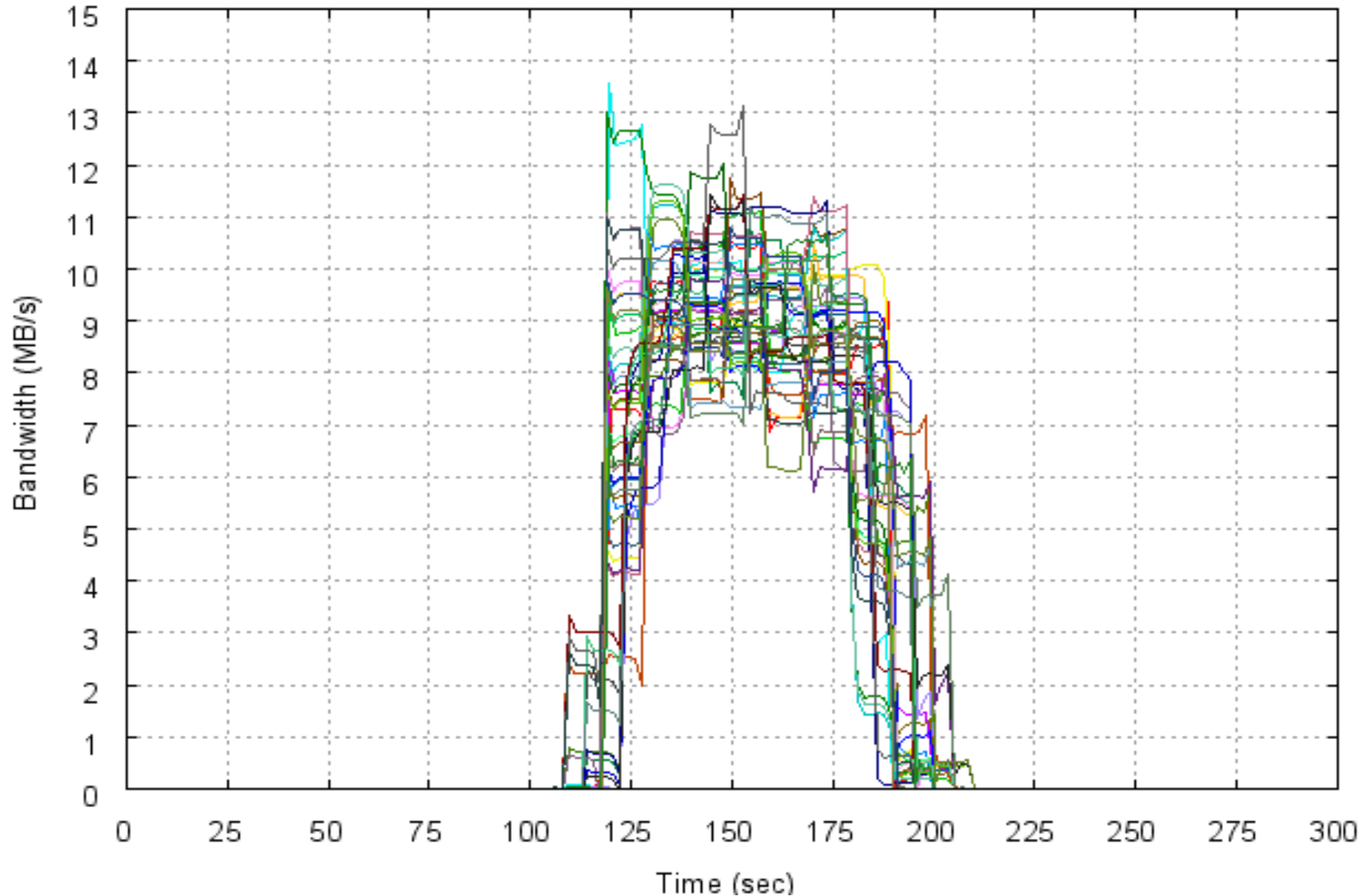


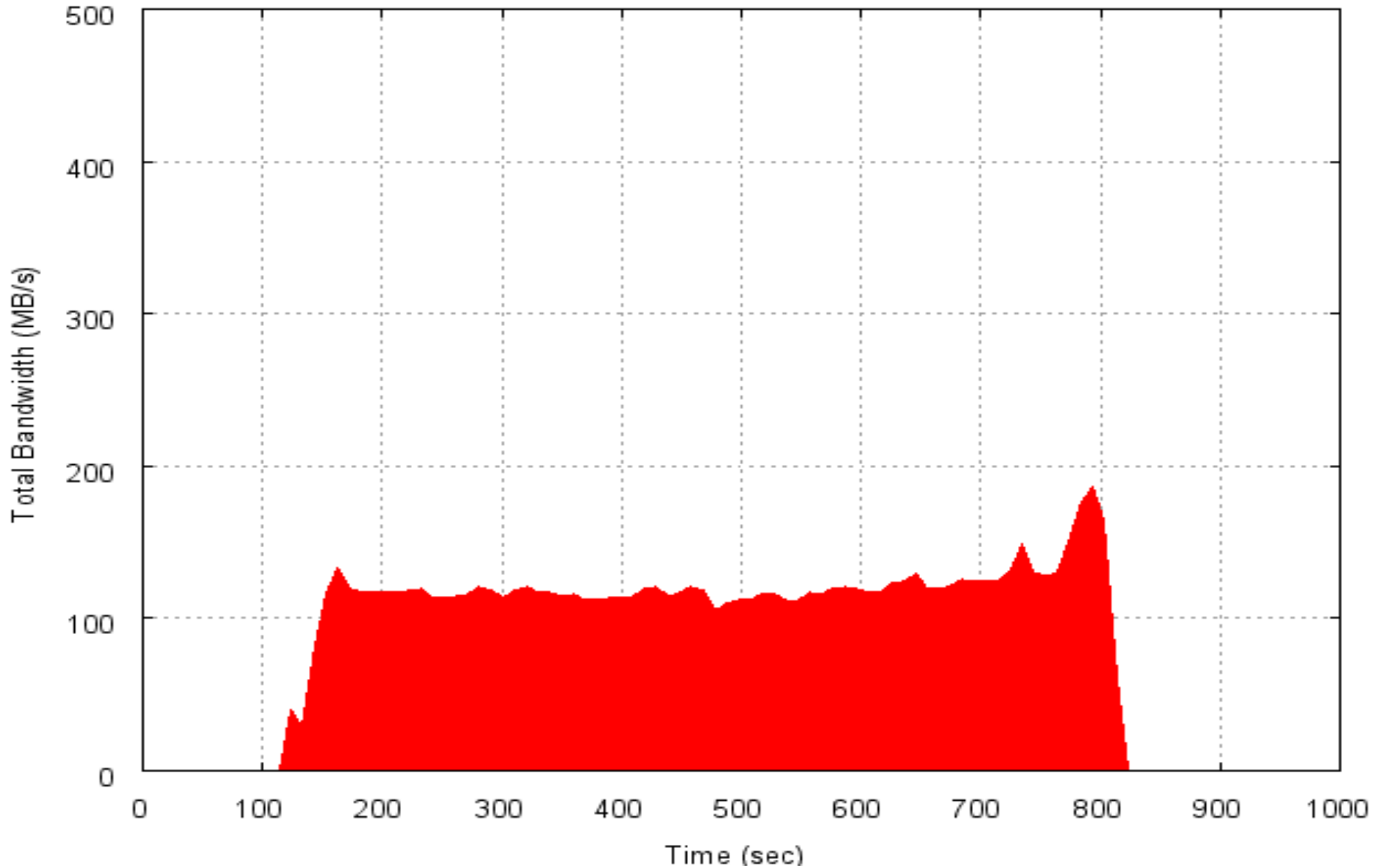
SystemImager (multicast transport): installation of 50 clients

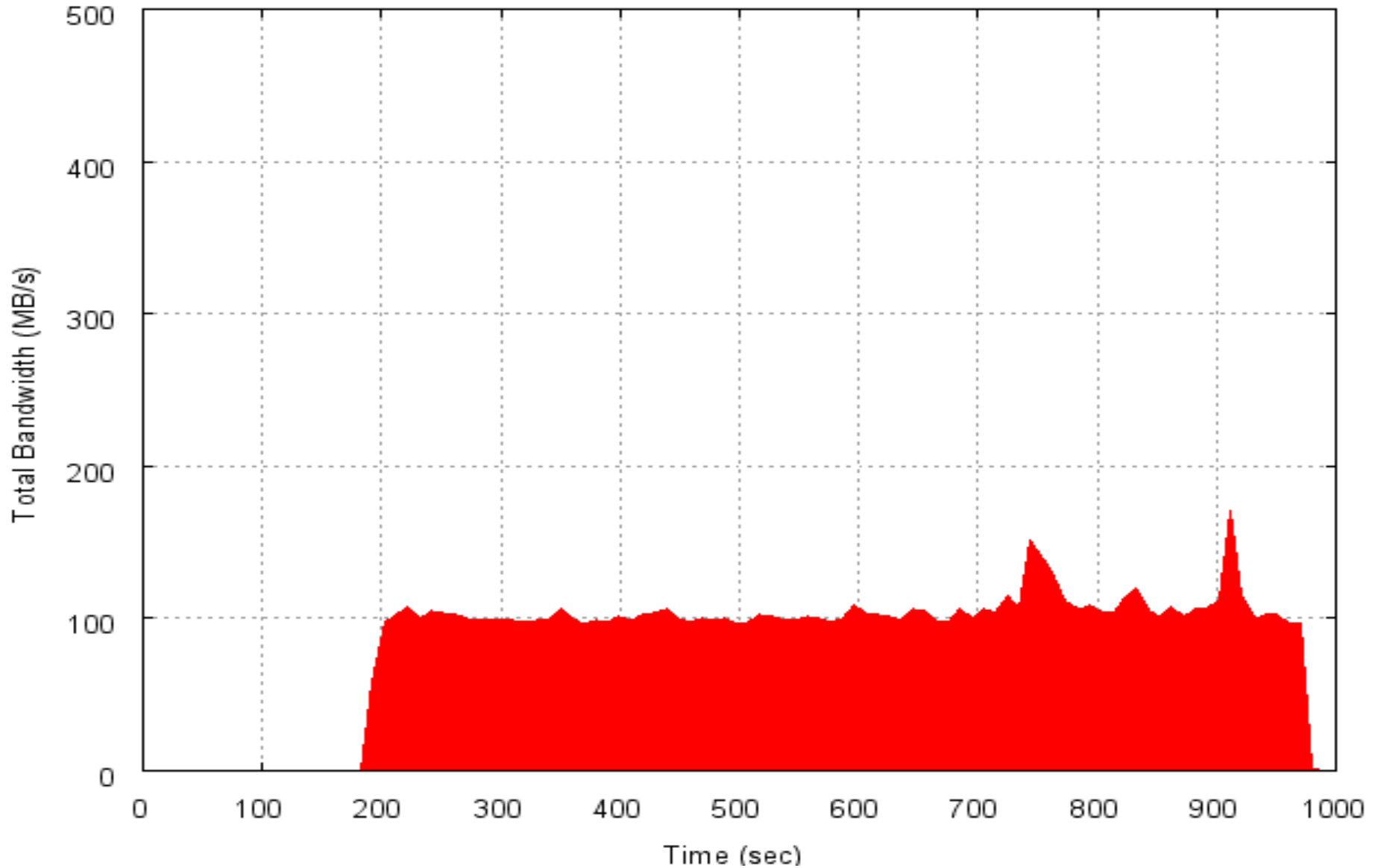


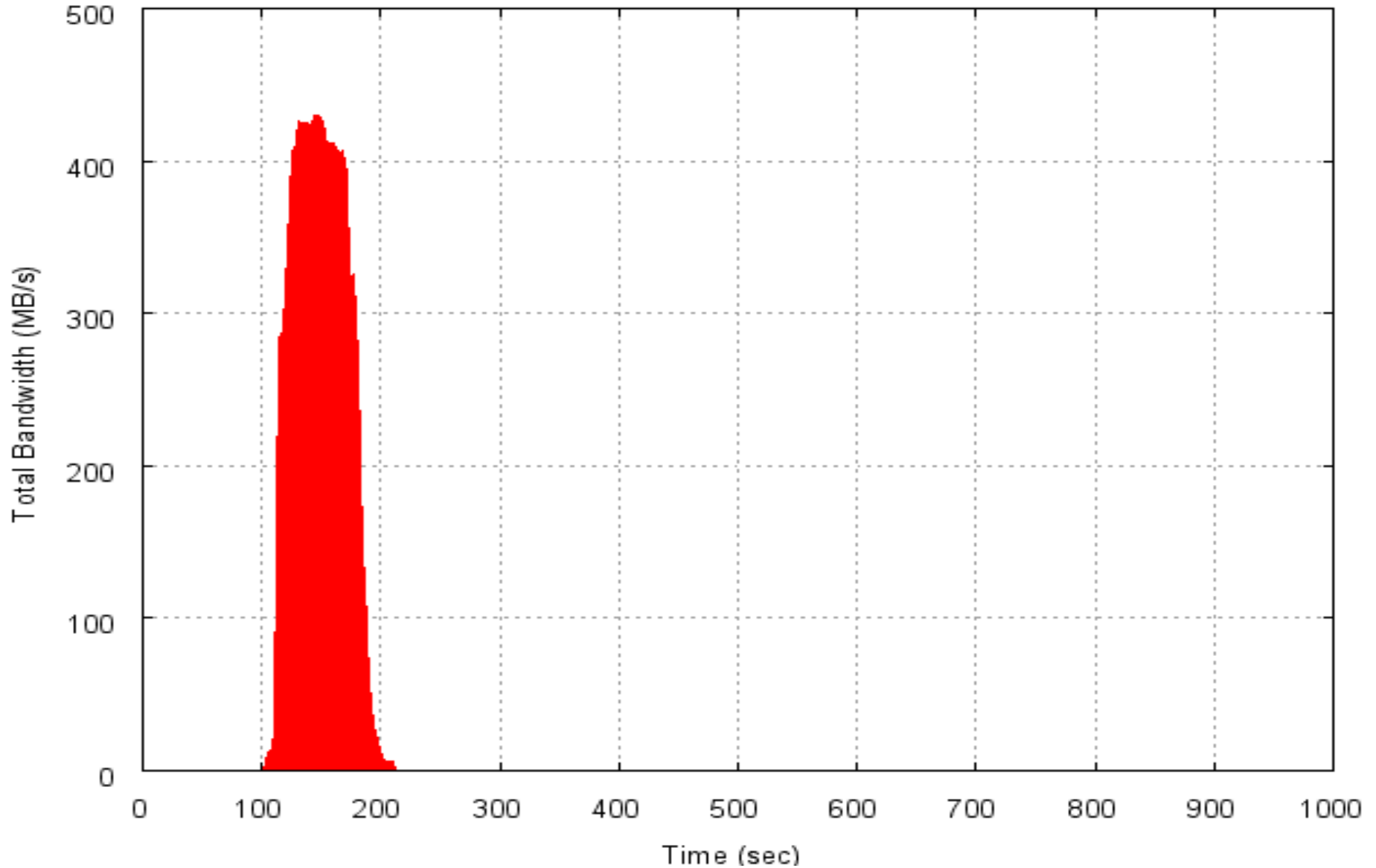
BitTorrent: 50 clients (download rate)

SystemImager (BitTorrent transport): installation of 50 clients











Conclusion

- Quicker deployment of images
- Safer deployment (better error handling)
- Less load on the image server
 - ✓ no need to buy a powerful machine

- More disk space consumption!
 - ✓ images + tarballs of images
- Images and tarballs must be kept in-sync
 - ✓ Re-generate tarball and .torrent at each image change

- Improve security (encryption of BT tarballs)
- Exploit the p2p approach to create distributed and redundant repositories of custom image
- Virtual cluster deployment (re-imaging using the same physical resource pool)
- Use BT transport also for updates (pushing changes/differences of images)

- Web:
 - ✓ <http://www.systemimager.org>
- Mailing list:
 - ✓ sisuite-users@lists.sourceforge.net
 - ✓ sisuite-devel@lists.sourceforge.net
- IRC:
 - ✓ [#sisuite](irc://irc.freenode.net/#sisuite) (irc.freenode.net)



Tank you for attending