Proposal for Research project openMosix

Maarten Michels & Wouter Borremans

December 15th, 2004

1 Preface

In this research project we will have a closer look to several aspects of openMosix[1]. openMosix is a Linux kernel extension for single-system image clustering which turns a network of ordinary computers into a supercomputer.
The project is carried out by:

- Maarten Michels[3]
- Wouter Borremans[4]

The project will be supervised by Harris Sunyoto[2]

Details about the project can be found at: `http://www.os3.nl/~wborremans/rp1.html`

2 Project research questions

In this project we will look at the following characteristics of openMosix:

- Performance
  - Is the performance linear?
  - What applications will be benefited by such a system?
- Reliability
  - What happens to the data if one node fails?
- Network load
  - How is the network load related to the number of nodes?

3 What will we use to carry out this project?

In this section we describe the technologies we will use to achieve our goals.

3.1 Computer infrastructure

The SNB lab offers us 15 computer which we will use to do the tests. The computers consist of a 1Ghz Pentium III processor and 256MB internal memory. One computer will act as a server, the others will be the clients of the cluster.

3.2 Network infrastructure

The computers are coupled to each other with an 100 Mbit ethernet switched (layer 2) network. The switch we use is a 3COM Superstack 3 3300TM.

3.3 Software infrastructure

Easy future expandability of the cluster is possible using a netboot image at the clients. This netboot image is located at a server.

4 How are we going to test the research questions

In this section we will describe the tests we will carry out to test the openMosix[1] cluster. It could however be that some programs don't run on OpenMosix due to some limitation. This will however be reported.

4.1 Performance

We are going to use several different kinds of programs to test the performance on an openMosix cluster. These are:

PovRay Povray[5] is an raytracing program. Ray-tracing is a rendering technique that calculates an image of a scene by simulating the way rays of light travel in the real world.

Compiling benchmarks for operating systems like Gentoo where everything needs to be compiled it would be nice to know if it would benefit from an OpenMosix cluster.

Encryption Benchmarks OpenSSL[6] is being used a lot and we where wondering if this could benefit from an openMosix cluster.

Synthetic Benchmarks Synthetic benchmarks show what a system should be doing.

During the test itself other programs encountered may also be used.

4.2 Reliability

We evaluate the reliability of the cluster by physically removing parts of the cluster, this means:

- Disconnecting nodes from network

  - Evaluate how failures of nodes are detected and handled (node data still available)

- Disconnecting power from node

  - Evaluate how complete node failures are detected and handled (node data not available)

4.3 Network load

We measure the network load using MRTG[7] and or CACTI[8]. These programs will generate graphs so we can analyse the network traffic. Both tools use the SNMP protocol to query the interfaces.

5 Project planning

- Week 1
  - Writing project proposal
  - Setting up the test environment
  - Looking for performance measuring applications
- Week 2-3
  - Performing predefined tests
- Week 4
  - Finishing loose ends
  - Writing project report

References

[1] openMosix, Bruce Knox
`http://www.openmosix.org/`

[2] Harris Sunyoto
`http://staff.science.uva.nl/~sunyoto/e-mail.html`

[3] Maarten Michels
`http://www.os3.nl/~mmichels`

[4] Wouter Borremans
http://www.os3.nl/~wborremans

[5] Povray, Vision Raytracer Pty. Ltd, http://www.povray.org

[6] OpenSSL, http://www.openssl.org

[7] MRTG,Tobias Oetiker oetiker@ee.ethz.ch andDaveRand-dlr@bungi.com, http://www.mrtg.org

[8] Cacti, http://sourceforge.net/projects/cacti/