International Institute for Applied Systems Analysis
Schlossplatz 1 • A-2361 Laxenburg • Austria
Telephone: (+43 2236) 807 342 • Fax: (+43 2236) 71313
E-mail: publications@iiasa.ac.at • Internet: www.iiasa.ac.at

Interim Report **IR-99-20**

World Wide Web Robot for Extreme Datamining with Swiss-Tx Supercomputers

Armin S. A. Roehrl (Armin.Roehrl@epfl.ch)
Swiss Institute of Technology, Lausanne, Switzerland.

Martin Frey (Martin.Frey@scs.ch)
Supercomputing Systems AG, Zurich, Switzerland.

R. Alexander Roehrl (Roehrl@iiasa.ac.at)
International Institute for Applied Systems Analysis, Laxenburg, Austria.

Approved by

Gordon J. MacDonald (macdon@iiasa.ac.at)
Director, IIASA

June 1999

Abstract

This paper discusses the software and hardware issues of designing a highly parallel robot for extreme datamining on the Internet. As a sample application, a World Wide Web server count experiment for Switzerland and Thailand is presented. Our platform of choice is the SwissTx, a supercomputer built from commodity components that runs NT and COMPAQ Tru64 Unix. Hardware and software of this machine are discussed and benchmark results presented. They show that NT is a feasible choice even under the given extreme conditions. Using statistical modelling for optimizing the search process, the inevitable bandwidth problem is reduced to some extent to a computation problem. We suggest that our approach to Web robots is a robust bet for a multitude of future Internet applications which might lead to a large-scale and cost-efficient usage of Web robots.

Contents

Acknowledgments

We would like to thank the CTI (Commission for Technology and Innovation) for sponsoring this project, R. Gruber for providing access to the latest hardware and the system administrators of domain zones, in particular the SwitchNG, for the permission to do full zone transfers. Finally, we are grateful to A. C. Davison for his support.

World Wide Web Robot for Extreme Datamining with Swiss-Tx Supercomputers

Armin S. A. Roehrl*, Martin Frey†, R. Alexander Roehrl‡

1 Introduction

This paper presents the concept and results of a first World Wide Web (WWW) server count experiment with a parallel Web robot that runs under NT on Swiss-Tx which are supercomputers built from commodity components. The robot is scalable in that it works in any heterogenous clusters that support a Java Virtual Machine (JVM)

The currently exponentially growing size of the Internet, in particular the size of the World Wide Web (WWW) is difficult to measure. To find the number of registered hosts, most experts use a method called *simple hostcount*, whereby one zone after the after is scanned. Apart from all registered hosts there are also be some unlisted machines that are hidden behind a firewall or that are momentarily not connected to the Internet. In hostcount, a certain percentage of machines gets *pinged*, i.e. a data packet is sent to a host and bounced back to see whether the machine is "alive". The hostcount in January 1999 estimated some $43,230,000$ hosts worldwide from which $8,426,000$ replied to a ping [10].

With increasing importance of the WWW, in particular in the context of E-commerce which is expected to hit 1 trillion US dollars by 2003 [11], the statistic of the number of Web servers is of growing interest. The concept of a Web robot [21] has the advantage of producing additional information as a virtually free by-product. For example, the Web robot collects information on the server software used, and many statistics such as for example on the use of the robot-exclusion standard, acceptance of Java programming language, copyright violations and of course the number of Web servers. The set-up of the authors' *Web robot "Ellen"* is an intelligent extension of that of robots used by some current search-engines to update their URL directories. In contrast to many brute force search schemes currently used on the WWW, Web robot Ellen has a built-in statistical confidence strategy to determine where and how long to search. This paper focuses on the computing aspects of the Web robot, for details about the statistical aspects of the search strategy, please see [21]. Optimal design properties for a software robot that operates on the distributed and non-symmetrical WWW are parallel slackness and a large bandwidth access to Internet backbones. The Swiss-Tx supercomputing project facilitated the authors to use a highly parallel approach for the Web robot Ellen. The batch system is used to automatically start parallel threads on the platforms with the lowest load. Web robot Ellen was run in concert on a Solaris cluster of Suns, on NT on the Swiss T0(dual), and on COMPAQ Tru64 UNIX on the Swiss T0. The Java programming language was used, as it guarantees the portability of the software code between many different platforms and

---

[*] Swiss Institute of Technology, Lausanne, Switzerland

[†] Supercomputing Systems AG, Zurich, Switzerland

[‡] International Institute for Applied Systems Analysis, Laxenburg, Austria

facilitates networking.

The first part of this paper gives an overview of the Swiss-Tx Project, which provides supercomputers built from commodity components (see Section 2.1). Sections 2.2 to 2.4 explain the batch and interactive access with LSF, principles of the Remote Store Architecture and summarizes some results of the Parkbench suite which was used for the acceptance of T0 and T0(dual). Section 3.1 describes the Web robots software design principles and Section 3.2 summarizes illustrative Web server count results obtained for the .ch (Switzerland) and .th (Thailand) domain. Finally, the paper is concluded with an outlook on further potential use of supercomputing, NT and software robots in general.

2 The Swiss-Tx Project

General-purpose supercomputers are often built with custom hard- and software. This is a good approach due to long time-to-market and a relatively small supercomputer market compared to the server- and workstation market. Therefore, in the past four years the supercomputer community has lost a few American manufacturers: Convex has been taken over by HP, Cray by SGI, Intel has stopped the business of massively parallel computers, Connection Machines and KSR have disappeared from the market. The recent developments in the commodity market makes it possible to build a supercomputer out of commodity parts. In the Swiss-Tx project researchers and developers from the Swiss Federal Institute of Technology in Lausanne (EPFL) and Zurich (ETHZ), from the Swiss Center for Scientific Computing in Manno (SCSC), and from Supercomputing Systems AG in Zurich (SCS), jointly develop communication hard- and software, and integrate software packages like debuggers, batch management systems, profilers etc. in order to develop, build and install a series of commodity based supercomputers until the year 2000 [1, 2, 3]. The first machine of this series, the Swiss-T0 was installed at EPFL in December 1997 [5]. The second machine, the Swiss-T0(Dual) [6] was installed in September 1998.

2.1 The Swiss-T0(Dual) Machine and Future Machines

The T0(Dual) machine consists of 8 dual CPU Alphaserver 1200 from COMPAQ running Windows NT Server (Figure 1). The machines are interconnected with a low-latency, high-bandwidth bus-based interconnect developed by SCS and ETHZ [8]. The communication over this interconnect is programmed with MPI. Besides the MPI interconnect a fast Ethernet is used as a service network. The main goal of the Swiss-T0(Dual) machine is to test Windows NT as an operating system for scientific computing in a computing center. A user validation and a per user accounting facility was needed. Users working at UNIX based desktops on the EPFL campus must be able to access the machine.

The remote access from UNIX based desktops was solved by adding a frontend machine to the cluster to which to users could connect with telnet. The Load Sharing Facility (LSF) from Platform Computing was used to access the NT cluster from the frontend host. It is planned to use an NT terminal server as frontend host later.

In June 1999 the Swiss-T1 will be installed at EPFL. This machine will use a switch based interconnect. The machine consists of 32 dual CPU COMPAQ Alphaserver and two front-end servers. The T1 will run COMPAQ Tru64 UNIX. The switch fabric for the MPI communication is based on 12 port, 100 MB/s bidirectional crossbar switches. The

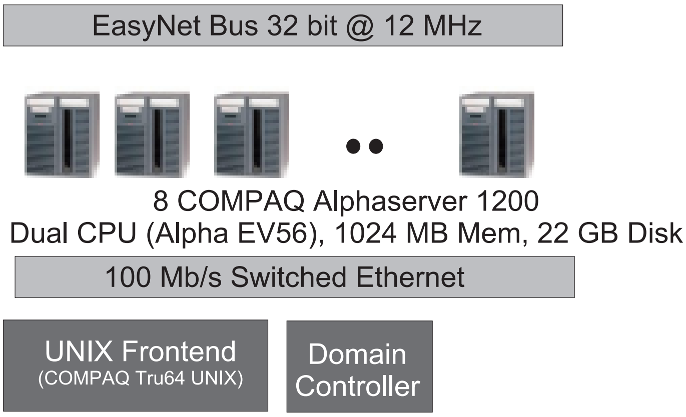

Figure 1: Configuration of T0(Dual) with the EasyNet bus, the Unix frontend and the domain controller

Alphaservers will be equipped with two PCI adapters in order to achieve a bidirectional bandwith of 200 MB/s. It is planned to install a supercomputer called Swiss-T2 with 500 processors by the year 2000.

2.2 Batch and Interactive Access with LSF

The user home directories reside on the Unix frontend. All compilations and computational runs were executed through the batch system of LSF. The home directories were exported to the NT machine using SAMBA. The user validation was done by LSF based on the Unix password mechanism. The user processes on the NT machines were running with the privileges of the user submitting the job. The job accounting (CPU usage, I/O usage) of LSF was used. LSF provides all the needed features to implement a multi-user, general-purpose supercomputer with Windows NT. The configuration is quite difficult for shared Unix-NT clusters and should be improved by Platform.

2.3 Communication Libraries

To achieve a low latency, high bandwidth communication the latency of operating system calls must be avoided. In the Remote Store Architecture the device driver maps the communication hardware into the memory space of the application. The application can directly access the communication hardware without any operating system interaction. By avoiding the operating system, the security mechanisms of the operating systems like parameter checking or resource protection are bypassed. The Remote Store Architecture hardware implements those mechanisms in hardware. The bus-based hardware allows only one application to access the EasyNet bus. The switched-based hardware will allow one application per PCI adapter. The structure of the communication libraries is shown in Figure 2. For a detailed description of the communication libraries see [4].

2.4 Benchmark Results for MPI

The MPI libraries were benchmarked with the communication low level benchmarks of the Parkbench suite [9]. The COMMS1 benchmark measures half round trip latency and the bandwidth with a ping-pong algorithm. A message is sent out by the first process, received by the second and sent back. As soon as the first process has received the message, the next message is sent out. The latency of the full round trip is measured and divided by

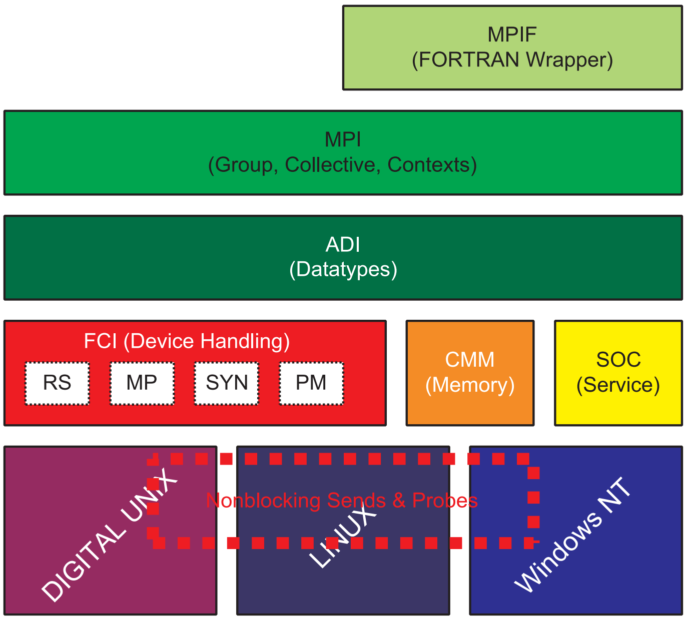

Figure 2: Structure of the communication library and device driver of the Remote Store Architecture

two. The results are shown in Figures 3 and 4. The results are compared between T0 (a UNIX system with an older revision of the communication hardware) and T0(Dual) (the NT system described in this paper). Figure 3 shows that a latency of about 10 μs was achieved. Figure 4 shows the bandwidth of 35 MB/s. The communication uses programmed I/O up to a message length of 1024 Bytes, for larger messages the DMA engine on the communication adapter is used. The setup of a DMA takes about 20 μs, but DMA transfers achieve a much higher bandwidth.

3 Application: Web Server Count

3.1 Design Principles of the Web Robot

Originally, Web robot "Ellen" was developed as a search-robot to build the URL data base for a search engine. To monitor the enormous growth of the WWW, the Web robot was developed further to determine the number of existing Web servers. A lower bound for the number of WWW servers is given by the number of registered domains worldwide, while an upper bound is the number of hosts on the Internet.

In principle, one could try to probe every single host (IP number) that is registered to identify Web servers. "Probing" means to send a *"GET"* (HTML protocol) request to the default port 80 of the host in question and to see whether the machine responds. This brute force method gets more and more infeasible as the size of the Web size continues to increase. However, to do such a search for a few country domains is a way to test the validity of strategies and statistical methods used to perform eventual global Web server counts. A number of pitfalls have to be avoided when doing brute force searches which may look like possible intrusion [12] to system administrators[1]. A practical way is to look at the top of every domain first: Suppose the domain is called *microsoft.com*,

---

[1] Surprisingly, after examining more than two million IP numbers that way we got only two mails from people that noticed our scan.

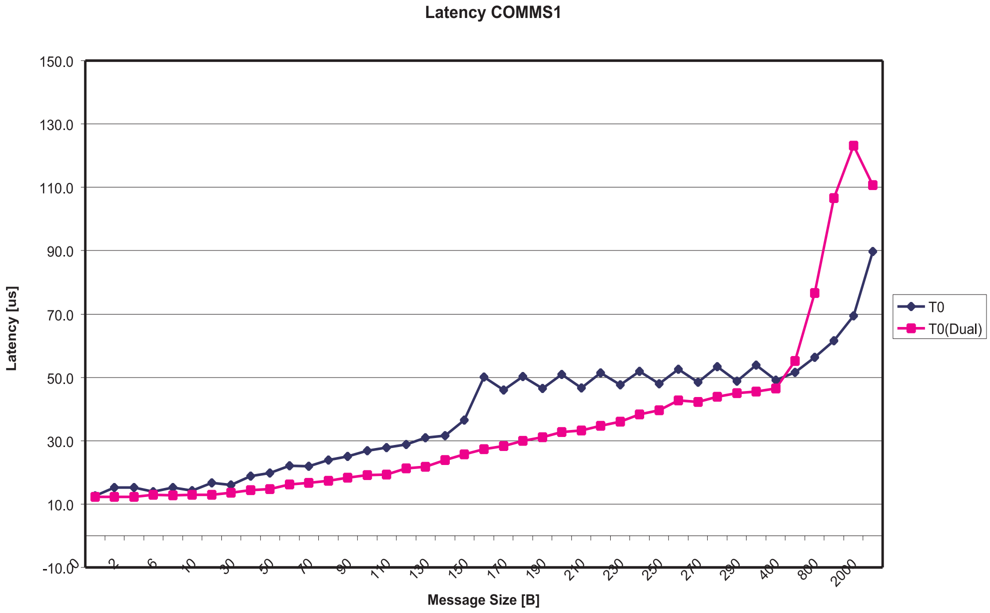

Figure 3: Latency measured with the COMMS1 benchmark (ping-pong).

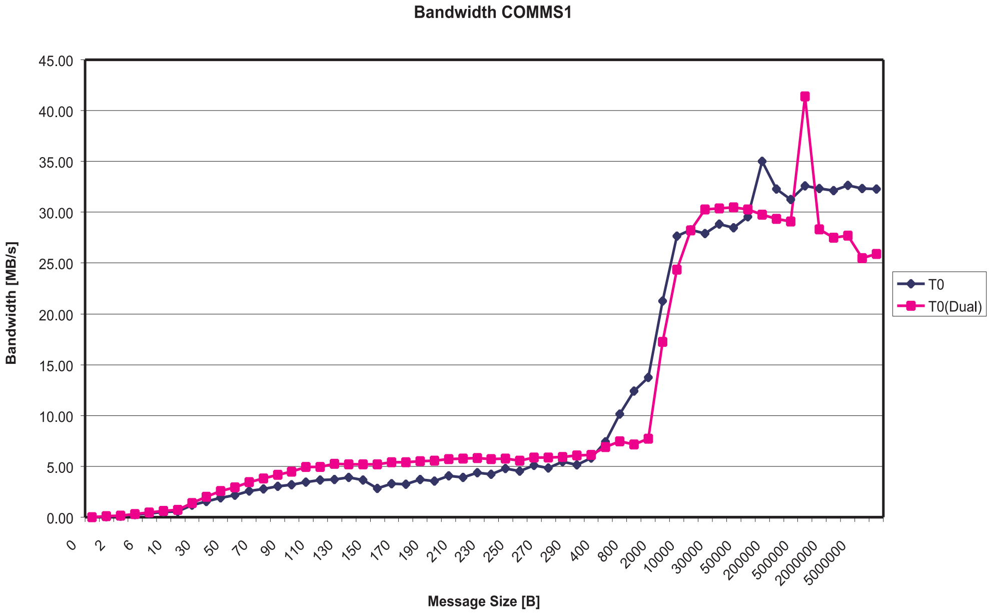

Figure 4: Bandwidth measured with the COMMS1 benchmark (ping-pong).

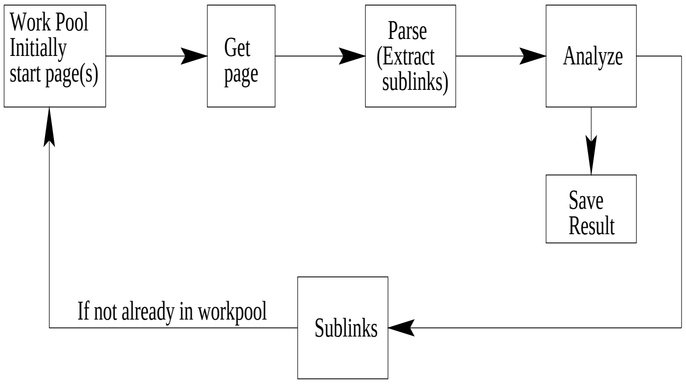

Figure 5: Simplified search scheme of the Web robot. It downloads a page, analyses it, stores the results and continues to process the links on the page, provided the analysis shows that it might be worthwile to continue with the links on the page.

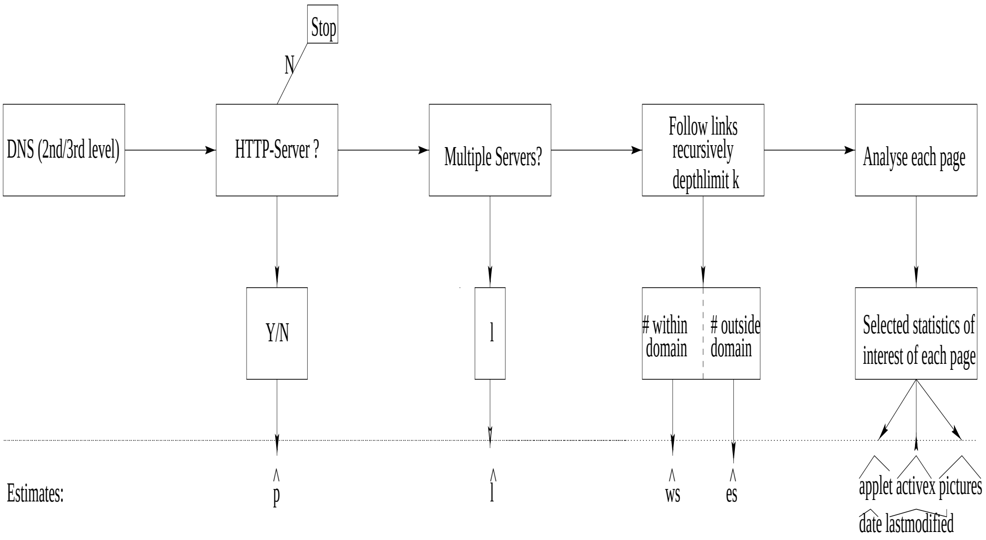

Figure 6: Hierarchical search structure of the Web server count: The Domain Name Servers provides the domain name to test. Then if lower in the hierarchy a Web server is existing, down to a certain depth links are followed recursively. Statistics of interest are continuously collected.

then first *www.microsoft.com* and *microsoft.com* is checked. Where possible the number of machines behind that top one is estimated. For example one finds that behind Microsoft's www.microsoft.com there are at least six machines. Finding out how many Web servers are behind the top server is difficult. The Web robot has to follow Web links recursively, down to a certain depth, always checking to avoid loops. The search depth is controlled by statistical methods. See Figure (5), (6) and (7). It requires the robot to read in the full HTML code of the pages. On the one hand this requires enormous computing resources, large bandwidth and stability of the system. On the other hand, it provides more interesting information than simple ping methods. Building a Web robot and launching it on a large scale is a difficult task. When a robot requests too many pages from one server, this is called rapid-fire and may take servers down, or at least make them very slow. Therefore, the parallel robot puts all newly found links from different servers into a workpool (see Fig. (5)) from which new links are randomly drawn and examined next.

"Stalling" messages that transform the WWW into *World Wide Waiting* are due to lack-

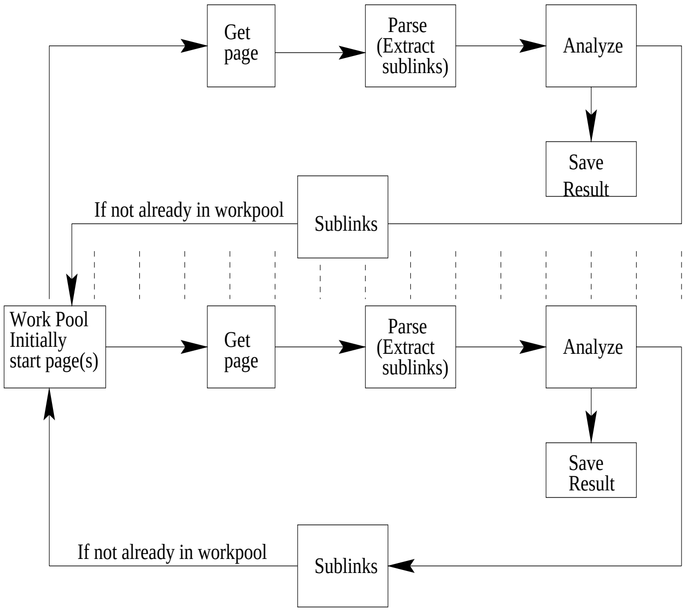

Figure 7: A multithreaded version of the Web robot. The dashed lines indiate additional threads running in parallel.

ing bandwidth or too little Webserver computing power as compared to the Web traffic. Web robot Ellen has a 40Mbps bandwith available with the USA and 100Mbps connection available to other Swiss universities and to SwitchNG [13]. This bandwidth is of the same order of magnitude as recently implemented among many US universities under the US "Next Generation Internet Initiative" [15]. However, this bandwidth is at least one order of magnitude smaller than that achieved in commercial Internet backbones deployed by leading companies (e.g. [17]) and 3+ orders of magnitude smaller than that achieved in current photonic testbeds in Europe [16], the US [18, 17] and Japan [19].

We use parallel-slackness, i.e. the robot opens many connections (page requests) simultaneously. While it is waiting for an answer from the server to its request, other requests can be processed at the same time. This is especially important as page download sizes and bandwidth availability vary greatly. Two levels of parallelism are overlapping: Parallelism on one processor (see Figure (7)) and parallelism by starting several threads and running them on different processors at the same time. The relatively low data dependency makes it an excellent task to run in parallel. The different search results are synchronized from time to time to decide whether to contine with the search process or to start a new thread.

3.2 Web Server Count Results

The emphasis of this paper is on the computer science aspects. For the relevant statistical modelling background and detailed host and server count results, the authors recommend a forthcoming paper [21] which compares traditional survey sampling approaches with more advanced Bootstrap solutions in a Bayesian context, where second level domains are treated as one strata.

Here a summary of Web server count results follows, see Table 8. The data is from a test run with the robot Ellen in October 1998. At that time, the official SwitchNG domain zone entry consisted of 80,875 and the Thailand domain of 1,931. The latter is split in

Statistic	Switzerland	Thailand
Servers up	89%	63%
Robots-exclusion standard	19%	11%
Most popular server	Microsoft (38%)	Apache(53%)
Second most popular server	Apache (33%)	Microsoft (25%)
Average page age	127 days	139 days

Figure 8: Overview of Web server count statistics. The number of servers "alive" is low, since some domain names are only reserved for later use. The robots-exclusion standard refers to setting up a "robot.txt" file to ban robots from certain sensitive parts of the Web space. "Most popular server" refers to server software.

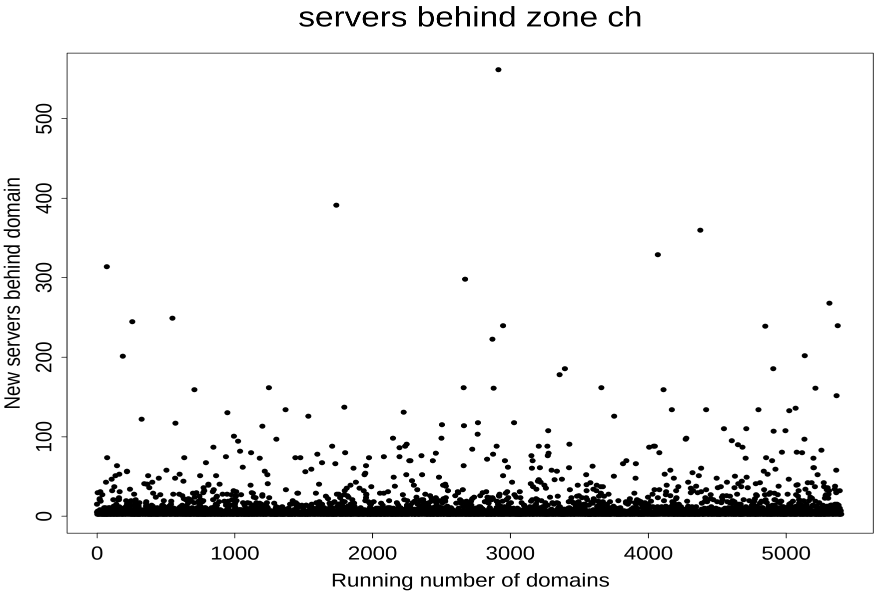

Figure 9: Domain .CH (Switzerland); x-axis: Running number of domains; y-axis: New servers behind domain.

eight major second level domain zones: 204 (ac.th), 148 (or.th), 1492 (co.th), 86 (go.th), 5 (mi.th), 19 (net.th) 0 (amazingthailand.th) and 0 (asiangames.th) [2].

Figure 8 shows a considerable share of servers that are not "alive". In part this has to do with the fact that some domain names are only reserved for later use. The larger share of commercial server software and increased use of robot-exclusion standards etc. in the Swiss domain, as compared to the Thai one, reflects the different phases of WWW development in the two countries. In Switzerland, the E-commerce sector is already of a much larger importance than in the still predominantly academic Thai WWW. However the surprisingly similar average age of the Web pages of around one third of a year is a simple consequence of the exponential growth of the WWW with a similar growth rate, although in a different state of diffusion depth, in both Switzerland and Thailand. Figures 9 and 10 present sample results of the number of servers discovered behind a selected set of domains in the .ch and the .th domain respectively. Note the relative high number of hosts with relatively few (less than 3) links to other servers in Figure 9 and 10. These are mainly servers from companies that are outside a firewall. For obvious reasons companies

---

[2] asiangames.th and amazingthailand.th are for national events. They are special domains which are rarely assigned and the domain is revoked after the event [14].

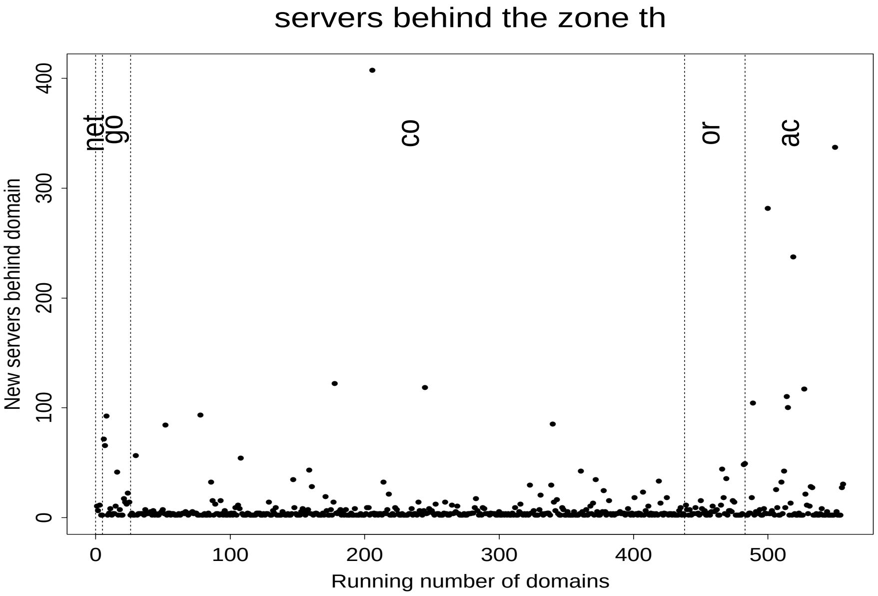

Figure 10: Domain .TH (Thailand); x-axis: Running number of domains; y-axis: New servers behind domain. The vertical dashed lines indicate from what second level domain the samples were taken.

do not link to their competitors. As shown in by Figure 9, cases with 100 or more servers behind the domain most likely belong to a University.

4 Outlook

This paper outlined the concept and results from a first test of WWW content analysis with a parallel Web robot. This extreme datamining of the Internet was only made possible by using state of the art hardware and good software design. This last section explores the conditions under which the outlined Web robot concept may become of widespread use.

One obvious constraint is bandwidth. Figure 11 shows the exponential growth of the commercial and the experimental transmission capacity in Gbit/s over time as compared to the increase in Microprocessor Performance. They show the same increase. However, there is a time lag of approximately seven years between commercial and experimental capacity. Estimated Internet traffic increase[3] significantly exceeds the performance increase of photonic and computing technologies. Because of the distributed, non-symmetrical nature of the WWW, the obvious bandwidth problem may be eased to some extent with a parallel approach that draws on considerable computing power, such as e.g. Swiss-Tx, as outlined above. Then in the years to come the use of Web robots and intelligent agents may become widespread, especially in business to provide a myriad of information relevant to e-commerce. These businesses will look for cost-efficient large computing resources such as future supercomputers built from commodity components running preferably under NT. In the authors' experiments, NT did not perform significantly worse or better than COMPAQ Tru64 Unix running Java on the T0dual. Java has proven that for certain types of applications it is mature enough to be used on a large scale application. The future of high

---

[3] MCI estimates a 30% Internet traffic increase per month [17].

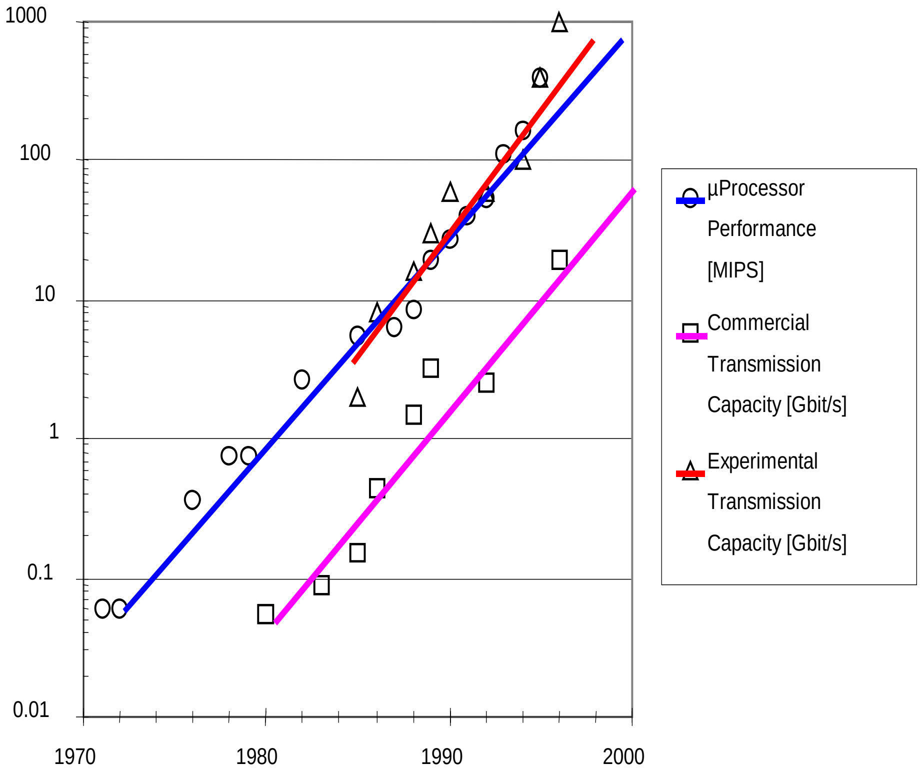

Figure 11: Photonic Technologies– Increase of performance with time (logarithmic scale) [22].

performance computing may be the cheap mass manufacturing of parallel machines built from commodity components. Established search engines like Inkomti use similar kind of hardware, e.g. clusters of Sun workstations. The uniqueness of the Swiss-Tx approach presented here is its switch.

As future agents and robots will become more popular and more widely used, the next challenge will be to make these different kinds of robots to communicate with each other. Possible applications range from Web robots acting as cybercops on copyright violations to increasingly sophisticated second and third generation search-engines.

References

[1] Gruber, R., Gunzinger, A.: The Swiss-Tx Supercomputer Project. EPFL Supercomputing Review, **9** (1997) 21–23

[2] Swiss-Tx Architecture, 1999. `http://capawww.epfl.ch/swiss-tx/`

[3] Gruber, R: Parallel Computer Architectures for Commodity Computing. Keynote speach at HPCN Europe '99, Amsterdam

[4] Brauss, S., Frey, M., Lienhard, M., Nemecek, J., Gunzinger, A.: Swiss-Tx Communication Libraries, HPCN Europe '99, Amsterdam

[5] Frey, M., Gruber, R., Kuonen, P.: Swiss-T0 Installation and Acceptance Report. (1998)

[6] El Mansouri, N., Gruber, R., Frey, M.: Swiss-T0(Dual) Installation Report (1998)

[7] Brauss, S., Nemecek, J.: The FCI Reference Manual, 1999. `http://www.ife.ee.ethz.ch/hpc/fci/`

[8] The Remote Store Architecture, 1999. `http://www.scs.ch/rsa.html`

[9] PARKBENCH (PARallel Kernels and BENCHmarks). `http://www.netlib.org/parkbench/`

[10] Internet Domain Survey, Network Wizards, January 1999, `http://www.nw.com/zone/WWW/report.html`

[11] E-COMMERCE: A trillion-dollar industry by 2003, ISOC Forum, Editor: Rickard, W., 08/01/1999

[12] Wunsch, J., Interface Business GmbH, Dresden, Personal communication;

[13] Despond, Y. , Swiss Federal Institute of Technology, Lausanne, Personal communication;

[14] Pensri, A., Personal Communication, Root of the .th domain zone server.

[15] Large Scale Networking Group of the CIC R&D Subcommittee, *The Next Generation Internet Initiative, a Draft Concept Paper*, US, 1997.

[16] European Commission DGXIII, ACTS (Advanced Communications Technologies and Services) Annual Programme Report, Brussels,1998. See also `http://www.cordis.lu`.

[17] Goldstein, S. N. , Approaching the Gigabit Networking Threshold in the US. In The International Symposium on Gigabit Networks, Singawa-ku Tokyo, 1997.

[18] US DARPA Projects in the Photonics Domain, 1998. `http://www.ito.darpa.mil`

[19] Oda, K. et al. , WDM Ring Network Experiment, in OECC'97, Seoul, 1997.

[20] Roehrl, R. A. , Towards Terabit Networks – Japan and the US with reference to European Commission activities, European Commission DGXIIIB Stage Report, Brussels, 1997.

[21] Roehrl, A. S. A. , Controlling a highly parallel robot for extreme datamining on the Internet, to be submitted to the Journal of Computational and Graphical Statistics.

[22] Intel Corporation, Contribution to ECOC Conference, Oslo, 1996.