Twenty-Plus Years of Netlib and NA-Net

Jack Dongarra, Gene Golub, Eric Grosse, Cleve Moler, Keith Moore

Abstract

The Netlib software repository was created in 1984 to facilitate quick distribution of public domain software routines for use in scientific computation. The Numerical Analysis Net (or "NA Net") had its roots in the same period, beginning as a simple file of contact information for numerical analysts and evolving into an email forwarding service for that community. It soon evolved to support a regular electronic mail newsletter, and eventually an online directory service. Both of these services are still in operation and enjoy wide use today. While they are and always have been distinct services, Netlib and NA-Net's histories are intertwined. This document examines the histories of both Netlib and NA-Net and their impact on their user communities.

1. Introduction

The original Netlib repositories were established around July of 1984 by Eric Grosse, of Bell Labs in Murray Hill, NJ; and Jack Dongarra, then of Argonne National Laboratory. Each repository made a collection of high-quality mathematical software available via electronic mail - the Bell Labs server provided access via UUCP protocols, while the Argonne server made the collection accessible via Internet mail. The Netlib collection quickly spread to other servers around the world. As Internet protocol networks became more ubiquitous, other access methods were added, and though the electronic mail interface is still supported, most access today is via the world wide web. Netlib was one of the first “Open Source” efforts and helped establish the trend of distribution of software via the network, and it remains popular today.

By the time Netlib was established, an "na list" (na for "numerical analysis") of email addresses of numerical analysis specialists had been maintained for several years by Gene Golub, then Chair of the Computer Science department at Stanford University. By 1983 this list was being used to provide an electronic mail forwarding service. Mail to na.lastname@su-score would be forewarded to the list member with that last name. An email broadcast facility was also provided: mail sent to "na@su-score" would be forwarded to everyone on the list. By February 1987 this broadcast facility had evolved into a moderated email "digest" which soon became a weekly electronic newsletter. A "white pages" database and a world wide web interface were eventually added resulting in the set of services provided by NA-Net today. The NA-Net remains a widely used and valuable resource for the Numerical Analysis community. The NA-Digest is one of the oldest electronic periodicals, and continues to grow steadily in popularity.

2. Netlib

The goals of Netlib were to facilitate quick, easy, and efficient on-demand access to useful public-domain computational software of interest to the scientific computing community. The mechanism chosen for distribution of this software was electronic mail.

There were initially two repositories: one at Bell Labs and the other at Argonne National Laboratory. The Bell Labs server provided access via UUCP protocols, while the Argonne server made the collection accessible via Internet mail. To request files from either server, a user would send one or more commands as message text to that server's email address. For instance, a message consisting of the text "send dqag from quadpack" sent to either research!Netlib (UUCP) or Netlib@anl-mcs (Internet) would result in a reply consisting of the "dqag" subroutine from the "quadpack" package, along with any additional routines needed to use that subroutine.

In today's world of pervasive high-speed Internet access, it can be difficult to appreciate the environment in which Netlib was hatched or Netlib's revolutionary nature. The ARPAnet had been in existence for several years and software was commonly shared over the ARPAnet's FTP protocol, but the ARPAnet served a limited community and the Internet protocols which were adopted by the ARPAnet in 1981 were only beginning to see widespread use outside the ARPAnet in the mid 1980s. Software could also be uploaded to or downloaded from "bulletin board systems" or BBSs but this could entail significant long-distance telephone charges to peruse distant servers.

At the time Netlib was introduced, much distribution of public domain software was accomplished by sending physical media such as magnetic tapes through the postal service. This was difficult not only because of the time and expense of handling physical media, but also because of the lack of any widely used standards for writing magnetic tapes. Each computing platform wrote tapes in its native format; and because of differences in word size, record size, and organization of the data on tape, reading foreign tapes could be quite difficult and labor intensive. By contrast, out of necessity, ARPAnet electronic mail systems standardized very early on a common message format to be used across all platforms; and UUCP mail used a similar format. This made electronic mail a superior medium for the exchange of source code, at least for small files, and Netlib made effective use of it.

Netlib appeared at a time when electronic mail was beginning to be widely available within its community of interest. By utilizing electronic mail to distribute software, Netlib allowed users to avoid lengthy delays and sometimes costly procedures for acquiring software routines via physical media. Because the service was provided at no charge and users could submit requests directly to the repositories, users were able to acquire software without administrative approval that might have been required to make formal requests to organizations providing the software, or to pay for writing and shipping a magnetic tape.

2.1 Collection Criteria and Contents

Netlib limited its scope to mathematical software and related information of interest to scientific computing. There was also an effort to limit the collection to software of demonstrated high quality. The fact that high quality public domain software packages such as LINPACK and EISPACK were available for Netlib's initial collection helped to set high standards for future additions to the collection. More generally, the numerical analysis community's tradition of producing robust software to address clearly-defined problems made it somewhat easier to establish a high quality library for scientific

computing than in some other areas. Another way in which Netlib maintained its quality was by having all contributions reviewed by an editorial board. The board provided some assurance of the quality and stability of the software as well as some assurance that the software was up-to-date.

The initial collection contained LINPACK, EISPACK, MINPACK, FNLIB, routines from the book *Computer Methods for Mathematical Computations* by Forsythe, Malcolm, and Moler, QUADPACK, as well as a collection of "golden oldies". Soon additional codes were added from ACM's Transactions on Mathematical Software (TOMS), as well as some benchmarking codes, a biharmonic solver, multiprecision arithmetic package, BLAS, etc.

2.2 Server locations and mirroring

The Bell Labs Netlib server went online January 3, 1984, and was used for software distribution within Bell Labs during the spring. Both the Bell Labs and Argonne servers were being used by the public by July of that year.

The Argonne server was physically moved to Oak Ridge National Laboratory in October of 1989, when Jack Dongarra moved to Tennessee. As the Netlib collection became more popular, it became advantageous to mirror the collection outside of North America; however, the number of official mirrors was deliberately kept small to minimize the differences between the servers. A European mirror was established in Oslo in December of 1989, and an Australian mirror also existed by 1989. In May of 1993 the original Sequent server that had been in service at Argonne was replaced by two SparcStation 2 machines - one at ORNL and the other at the University of Tennessee in Knoxville. These servers were closely synchronized to one another, and both of the servers were configured to accept email requests sent to Netlib@ornl.gov domain. The ORNL server was retired in 1997. The official servers today consist of Netlib.bell-labs.com in Murray Hill, New Jersey; Netlib.org, at the University of Tennessee; the UK Mirror Service in Kent; and Netlib.no in Bergen, Norway. These servers are synchronized to one another via a process described in [1].

Since the introduction of anonymous FTP access to the primary Netlib servers many more unofficial mirrors have been created; and there is a great deal of variation among these mirrors with regard to the organization and currency of their collections.

There are a number of additional sites which mirror Netlib. We don't check on the following mirrors as closely.

- Netlib.uow.edu.au, Australia [ftp]
- AARNet (Australia/New Zealand only) [ftp]
- unicamp.br, Brasil
- Netlib.amss.ac.cn, China [ftp]
- Daresbury Lab, England

- Codiciel, France
- ENSEEIHT, France
- ZIB, Germany
- Athens, Greece
- Univ Thrace, Greece
- CASPUR, Italy
- C.D.S., Italy
- CILEA, Italy
- Phase AIST, Japan
- Ewha w. Univ., Korea
- ChgNet, Russia
- NCHC, Taiwan
- freesoftware.com, USA west

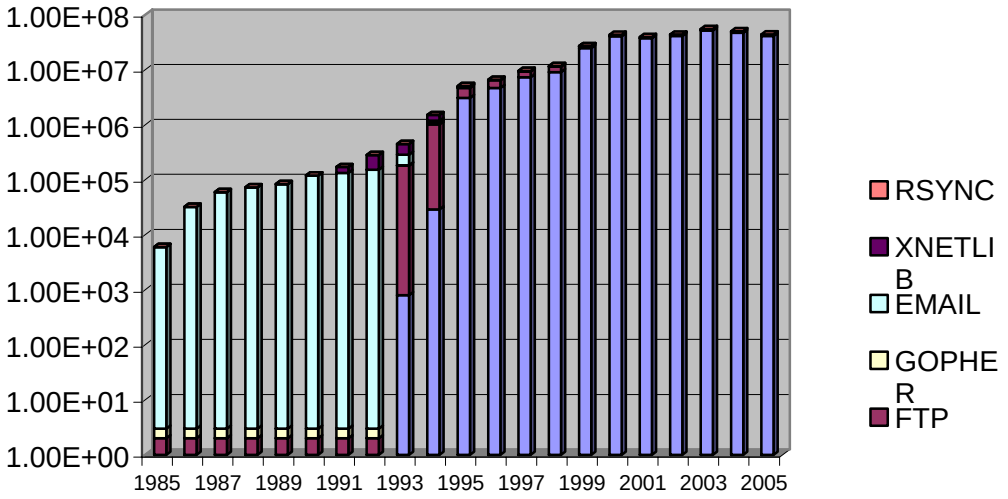
2.3 Access protocols

At the time Netlib was created there was no ubiquitous Internet, but rather an admixture of incompatible networks, each with different hosts, services, and means of addressing. In many cases the networks were interconnected with mail gateways but these could be difficult to use (due to the need to explicitly route a message through the gateways) and unreliable. As text-based electronic mail was the only service common to all of those networks, this was the means initially utilized by Netlib to provide access to items in its collection, but as Internet protocols began to be more widely deployed other methods which were easier to use became more feasible. A server for the Internet's File Transfer Protocol (FTP) was provided on the Bell Labs Netlib server in August 1991 and soon afterward at the ORNL server. The xNetlib browser was first deployed in December 1991, allowing users to peruse and download items from the ORNL server via a graphical user interface for the X Window System [2]. In response to the growing popularity of the World Wide Web, HTTP Netlib servers were established in October 1993 at UTK and ORNL; Gopher was enabled in November 1993. Although the email interface is still supported, HTTP and to a lesser extent FTP quickly eclipsed all other sources of Netlib traffic.

2.4 Usage Patterns

At least at the UTK and ORNL servers, Netlib usage rose steadily in the period from 1985-2000. Traffic levels have varied in recent years. Some of the variation in recent years may be due to an increased use of crawlers and mirrors. Approximately 80% of the

current traffic appears to be from individual users, 7% from mirrors, and 12% from web crawlers, which are presumably building indices for use by search engines.



Netlib Usage at UTK/ORNL
Figure 1

2.5 Community Impact

A large amount of mathematical software is both commercially and freely available. However, not all the software that is available is of high quality. It can also be difficult to locate the appropriate software by using web search engines, since the descriptions available for searching may be lacking or may not match the vocabulary used by the searcher. A good solution to these problems is to have experts in the field of numerical analysis maintain a moderated collection of high quality software which is organized and catalogued with appropriate metadata to enable easy searching. The Netlib (http://www.Netlib.org/) mathematical software repository is such a collection which has been contributed to and managed by the numerical analysis community for the past twenty some years. Netlib is continually evolving as new state-of-the-art software continues to be contributed by leading researchers in the field of scientific computing.

Netlib has been instrumental in promoting software reuse within the computational science and mathematics communities. Netlib has pioneered the idea of network delivered software, and has remained at the forefront of delivery technologies. The evolution of Netlib tracked the emergence of every important Internet application protocol, from email through FTP and HTTP. Additionally, Netlib showed that it is possible to manage such collections in a distributed manner, both in terms of infrastructure and administration. The Netlib collection itself has fallen behind the times, though much of it remains state-of-the-art. The fact that it still exists at all is some testament to good foundations, given the limited ongoing maintenance.

3. NA-Net

The Numerical Analysis Net (NA-Net) is a collection of several services designed to foster information and a sense of community among Numerical Analysts. It currently consists of an email forwarding service, the NA-Digest, and a whitepages database.

3.1. Software and Hosting

NA-Net traces its origins to a list of email addresses maintained by Gene Golub at Stanford University and distributed to others within the numerical analysis community. At some point an electronic mail system was specially configured so that mail to na.lastname@su-score (later, score.stanford.edu) would be forwarded to the person on the list with that last name. This was originally implemented by manually copying entries from Gene Golub's list into a system alias file (requiring interaction by the system administrator each time the list was updated). Eventually software was written by Mark Kent and Ray Tuminaro to allow the forwarding aliases and address list (for human perusal) to be generated from a common database [3].

One of the primary benefits of NA-Net's forwarding facility in those days was that it provided all of its subscribers with a uniform address. Before Internet access became ubiquitous, email traveled over a hodgepodge of dissimilar networks, each with its own addressing scheme. When sending a message between dissimilar networks, it was often necessary to "source route" the message through a gateway by embedding the recipient's address inside the address of the gateway. Because each network had its own addressing convention, an address that worked to reach a recipient from one location of the network would not necessarily work from another. But when sending mail to any NA-Net subscriber it was only needed to understand how to send mail to one na.lastname address; the same pattern would work for any other NA-Net subscriber. This took out some of the guesswork of mailing between networks, at least when mailing to other NA-Net subscribers.

A broadcast facility was also set up - so that a message to na@score.stanford.edu would be forwarded to everyone on the list. Eventually traffic volume and accidental misuse became significant enough that some sort of moderation was required for the broadcast facility, so it was converted to an email "digest". A moderator would review messages sent to na@score.stanford.edu, and the selected messages would then be sent out to everyone. The first issue of the digest was February 13, 1987. At first the digest was sent out at irregular intervals but soon settled into a weekly publication.

In December 1990 the NA-Net was moved to Oak Ridge National Laboratory, using new software written by Bill Rosener. The new software preserved the na.lastname forwarding and digest functions, but also allowed individuals to add themselves to the NA-Net, remove themselves, or change addresses - in contrast to previous versions which required manual maintenance of the subscriber list. To subscribe, a user sent an email message to na.join@na-net.ornl.gov with the following fields in the message body:

Firstname: *user's first name*
Lastname: *user's last name*

E-mail: *user's email address*

The na-net server would then reply with a message indicating whether the user was successfully added. To unsubscribe a user would send a similar message to the address na.remove@na-net.ornl.gov; to change an address the message was sent to na.change@na-net.ornl.gov. Each function had its own email address and its own requirements for the format of the data to be supplied. [4]

In May 1991 a "white pages" facility was added to NA-Net. Members could store information about themselves in the white pages database, such as their interests and home and work addresses. This information would then be made available in response to queries sent by email to na.whois@na-net.ornl.gov.

By June of 1993 the service had become so popular that the server was having difficulty handling the email traffic. At this time the entire NA-Net software package was entirely rewritten by Keith Moore to improve scalability (especially of email distribution) and robustness, but the user interface remained the same as before. This software remains in use today, with only minor changes. In November 1994 a web interface was added to rid users of the burden of having to submit requests in text-based email with rigidly defined syntax.

3.2 NA-Digest History and Content

Gene Golub was the original editor of the NA-Digest. In July of 1987 Gene Golub's began a sabbatical leave from Stanford and at that time Cleve Moler of MathWorks began editing the digest. With only occasional absences, Cleve continued to edit the digest until September of 2005. The digest is currently edited by Tamara Kolda of Sandia National Labs.

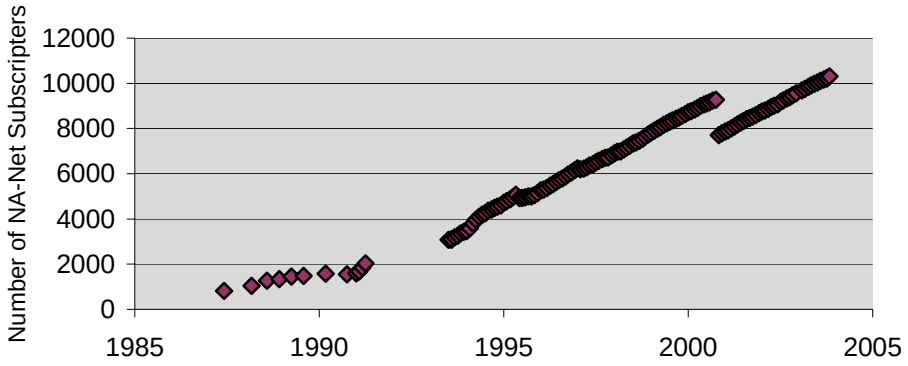
As the name suggests, the intent of the NA-Digest has been to have short announcements summarizing more extensive material available elsewhere. Today, almost all of the Digest contributions have URLs pointing to more complete announcements available on the Web.

The digest has generally contained anything of interest to the numerical analysis and mathematical software community. This has included both technical discussions and information about members of the community. Examples of material commonly appearing in the digest include: conference announcements, workshop announcements, advertisements for jobs, software release announcements, change of address announcements, new book announcements, journal contents, and notices about community members: awards, significant achievements, and deaths.

Reasonably complete archives of the digest exist and are hosted at http://www.Netlib.org/na-digest-html/. The archives contain a great deal of material of interest to anyone studying the history of numerical analysis software.

3.3 Usage patterns

Based on early reports in the NA-Digest and on log files maintained since 1993, the number of NA-Net subscribers has increased steadily from 821 subscribers in May 1997 to 11295 subscribers today (Figure 2). (The discontinuity in 2000 was caused by removal of many addresses that were no longer reachable, i.e. `BITNET` addresses.)



NA-Net Subscriber Growth
Figure 2

Several thousand messages per day are forwarded through na-net's email forwarding service. From a perusal of log file entries, many of these unfortunately appear to be spam. For many years the NA-Digest subscriber list was made available for download over Netlib, a holdover from the days when the network was small and most people with network access could be assumed to act reasonably. The list was also available to anyone by sending mail to na.sendlist@na-net.ornl.gov. Even after the list was removed from Netlib and the sendlist address was disabled, subscriber addresses were occasionally found to have been "leaked" by various means.

3.4 Community Impact

The international numerical analysis community is small enough that is still has a cohesive, “family” feeling. We believe that the NA digest has helped maintain that feeling. The fact that it is still low tech, simple text email means that it is accessible to anyone who has access to a computer and a network connection. This is particularly important to subscribers around the world who are unable to travel to many meetings. Moler reports that he has several times met people who recognize his name primarily as the “guy who sends me email every week.”

4. Related Projects

4.1 XNetlib

XNetlib began in 1990 and predates the web and tools like Netscape. XNetlib was a tool for "web based" software distribution. Whereas Netlib originally used e-mail as the user interface to the collection of public-domain mathematical software, XNetlib used an X-Window interface and socket-based communication. A user would download and installed the X-Window interface on their system, similar to NetScape, however XNetlib only could connect to the Netlib server. XNetlib made it easy to search through a large

distributed collection of software and to retrieve requested software in seconds. The last release of XNetlib was version 1.3. Most of the capabilities once provided by Xnetlib are now available using World Wide Web browsers.

4.2 NHSE

From 1994 - 2004, the NHSE (National HPCC Software Exchange) existed as a distributed collection of software, documents, data, and information of interest to the high performance and parallel computing community. The significance of the collaborative effort is evident through the many useful reports and tools generated as well as the many repositories that have been created, and are still being created, with the Repository in a Box (RIB) toolkit developed in 1996. However, continued operation of the site without funding has become impractical. Therefore, the site has been taken down. The NHSE meta-repostiory, which consists of metadata describing software applications and tools from the PTLib, HPC-Netlib and BenchWeb repositories combined, is still available. However, since PTLib and HPC-Netlib are no longer maintained, the metadata from those repositories are frozen in time. Only the BenchWeb content is still maintained.

4.3 RIB

Repository in a Box (RIB) provides a toolkit for building and maintaining metadata repositories. RIB was developed by the National HPCC Software Exchange (NHSE) Technical Team at the University of Tennessee, Knoxville. Initially, RIB only provided tools for the creation of software repositories. The recent release of RIB v2.0 allows RIB to create general metadata repositories. The creation of software metadata repositories remains the primary application of RIB. [5]

RIB has two primary design goals: promotion of software reuse and interoperability.

RIB promotes software reuse by providing tools to build metadata repositories. These repositories contain information pertaining to software packages and routines; an abstract, licensing information, point of contact, etc. These repositories are intended to be discipline oriented and are meant to act as a central access point for software information. The interoperability features of RIB allow repositories to share information in a scalable and efficient manner. Additionally, these features allow domain-specific repositories to be gathered into larger repositories with a common access point. For example, NHSE represents an aggregation of several other repositories. NHSE provides a common access point to these repositories - so far as the user is concerned, all the data is contained within NHSE.

4.4 NetBuild

NetBuild is a tool which automates the process of selecting, locating, downloading, configuring, and installing computational software libraries from over the Internet. Additional tools aid in the construction and cataloging of libraries in the format used by NetBuild.

NetBuild is easy to use: the user simply types "nb” followed by whatever command he would user to compile or link his program. NetBuild works by invoking the user's

command in an environment that intercepts calls to compilers and linkers. When NetBuild intercepts those calls it identifies and downloads any libraries that are needed, and passes the filenames of those downloaded libraries to the real compilers and linkers. Unlike many other tools, NetBuild is designed to work across a wide variety of computing platforms. In order to support high-performance computing applications, NetBuild can also perform fine-grained matching of libraries and target platform characteristics to select the version of a library that will provide the best performance for a given target platform[6].

5. Summary and conclusions

Software distributed by Netlib comes with the disclaimer that ``anything free comes with no guarantee". In contrast to commercial vendors like NAG and IMSL, Netlib offers no support beyond whatever documentation contributing authors choose to provide with their code. On the other hand, Netlib provides free, easy access to a large body of high-quality code, and the phenomenal growth of Netlib attests to the value of this service. We hope that Netlib, by making high-quality code even more accessible, will encourage software developers to make their source codes freely available and will make good programming easier for the scientific computing community.

For more than two decades, these characteristics have made Netlib a pivotal asset for the computational science community. The experience gained from Netlib has been transferred to a number of other projects designed to promote software collections management and software reuse. The National High Performance Software Exchange (NHSE) strove to apply the techniques and technologies of Netlib to more loosely coupled software repositories. The NHSE also attempted (and succeeded) in making software repository creation and maintenance easier through the Repository In a Box (RIB) toolkit.

Netlib and the NA-Digest have provided a singular resource for the advancement of science and education by making high-quality numerical software and interchange readily available to application scientists. It is well known that the use of high-quality software libraries can speed the development of new software applications. By making these libraries available to the science and engineering communities, Netlib has enabled a faster pace of innovation and investigation.

Additionally, Netlib-provided services such as NA-Net give members of these communities a forum for discussion and a means of maintaining contact with their peers. NA-Net in particular has also allowed for cross-pollination of ideas and techniques amongst scientists, engineers, and numerical analysts.

The idea that computational modeling and simulation represents a new branch of scientific methodology, alongside theory and experimentation, was introduced about two decades ago. It has since come to symbolize the enthusiasm and sense of importance that people in our community feel for the work they are doing. But when we try to assess how much progress we have made and where things stand along the developmental path for this new “third pillar of science,” recalling some history about the development of the other pillars can help keep things in perspective. It seems clear that while Computational

Science has had many remarkable youthful successes, it is still at a very early stage in its growth.

Many of us today who want to hasten that growth believe that the most progressive steps in that direction require much more community focus and funding on the vital core of Computational Science: *software and the mathematical models and algorithms it encodes.* But when it comes to advancing the cause computational modeling and simulation as a new part of the scientific method, there is no doubt that the complex software "ecosystem" it requires must take its place on the center stage.

At the application level the science has to be captured in mathematical models, which in turn are expressed algorithmically and ultimately encoded as software. Accordingly, on typical projects the majority of the funding goes to support this translation process that starts with scientific ideas and ends with executable software, and which over its course requires intimate collaboration among domain scientists, computer scientists and applied mathematicians. This process also relies on a large infrastructure of mathematical libraries, protocols and system software that has taken years to build up and that must be maintained, ported, and enhanced for many years to come if the value of the application codes that depend on it are to be preserved and extended. The software that encapsulates all this time, energy and thought, routinely outlasts (usually by years, sometimes by decades) the hardware it was originally designed to run on, as well as the individuals who designed and developed it.

Thus the life of Computational Science revolves around a multifaceted software ecosystem. But today there is (and should be) a real concern that this ecosystem of Computational Science, with all its complexities, is not ready for the major challenges that will soon confront the field. Domain scientists now want to create much larger, multi-dimensional applications in which a variety of previously independent models are coupled together, or even fully integrated. They hope to be able to run these applications on Petascale systems with tens of thousands of processors, to extract all performance as these platforms can deliver, to recover automatically from the processor failures that regularly occur at this scale, and to do all this without sacrificing good programmability. This vision of what Computational Science wants to become contains numerous unsolved and exciting problems for the software research community. Unfortunately, it also highlights aspects of the current software environment that are either immature or under funded or both.

Advancing to the next stage of growth for computational simulation and modeling will require us to solve basic research problems in Computer Science and Applied Mathematics at the same time as we create and promulgate a new paradigm for the development of scientific software. The efforts of Netlib and the NA-Digest, for the most part, have been through a volunteer effort. These services have been viewed as important and very successful. So the obvious questions come up. How should this kind of activity be supported? Where should it be done? Universities? Companies like MathWorks? We've been told many times by the funding agencies that "it isn't research". These are hard, but important, questions going forward. To make progress on these fronts will require a level of sustained funding from governmental sources to promote efforts like Netlib which is so fundamental to scientific research.

6. Acknowledgements

The authors would like to thank Don Fike for assistance with Netlib statistics, and Mark Crispin and Mark Kent for information about early NA-Net implementation and Mel Ciment who was at NSF and provided seed funding for Netlib and NA-Net when they first started.

7. References

[1] Eric Grosse. Repository Mirroring. *Transactions on Mathematical Software* 21(1), March 1995.

[2] Jack Dongarra, Thomas Rowan, Reed Wade. Software Distribution Using XNETLIB. *Transactions on Mathematical Software* 21(1), March 1995.

[3] Mark Kent. The Numerical Analysis Net (NA-NET). Technical Report 85, Institut für Informatik, Eidgenössische Technische Hochschule Zürich, January 1988.

[4] Jack Dongarra, Bill Rosener. NA-Net / Numerical Analysis Net. Technical Report ORNL/TM-11986, Oak Ridge National Laboratories, December 1991.

[5] S. Browne, P. McMahan, and S. Wells, "Repository In a Box Toolkit for Software and Resource Sharing," University of Tennessee, Knoxville, TN, Computer Science Technical Report, UT-CS-99-424, 1999. http://www.nhse.org/RIB/pubs/ut-cs-99-424.ps.gz

[6] K. Moore and J. Dongarra, "NetBuild (version 0.02)," Department of Computer Science, University of Tennessee, Knoxville, TN, CS Technical Report, UT-CS-01-461, March 2001. http://www.cs.utk.edu/~library/TechReports/2001/ut-cs-01-461.ps.Z