A Guide to Available Mathematical Software

By Ronald F. Boisvert

We have come to rely on a wide variety of tools to help us mine large research libraries for useful nuggets of information. Among these tools are cross-indexes, such as the card catalog, Mathematical Reviews, Current Index to Statistics, the ACM Guide to Computing Literature, and Science Citation Index, and the various classification systems on which these indexes are based. Today, we are seeing the explosive growth of the so-called information superhighway, a network-based resource that will far surpass any existing library in size and complexity. The networks, however, still lack many of the navigational aids common in library settings.

As a concrete example, consider an average computer user who would like to find software for solving a Sturm–Liouville eigenvalue problem. There are a number of good software packages for solving such problems. These packages might be installed in commercial libraries set up for use at a local computing site, or they might be research-grade public-domain packages available for downloading on the Internet. (Network-based distribution of such mathematical software was pioneered by the *netlib* project [4], and substantial collections are accessible in this way today.) Unfortunately, our user has a difficult time dealing with this bounty. Even for users who know where and how to traverse the labyrinthine computer network to a software repository, determining what is actually applicable to the problem at hand is still a tedious and error-prone process. Each repository may have a different access mechanism, and directory services are usually primitive at best. Our user would be greatly helped by an easily accessible, centralized information resource (i.e., a cross-index) with problem-oriented directory services and facilities for retrieving software and related items once they are located. Fortunately, systems with these characteristics are beginning to emerge.

Math software cross-index

The Guide to Available Mathematical Software (GAMS) is one such service. GAMS currently contains information on some 9000 problem-solving software modules from about 80 packages found in four physically distributed software repositories (three at the National Institute for Standards and Technology, and *netlib*). The system supports software searches by problem classification, keyword, or name, as well as retrieval of such items as abstracts, documentation, examples, and source code. (Some 32,000 objects can be retrieved in this way.)

GAMS performs the function of an interrepository and interpackage cross-index, as well as the functions of a repository itself (i.e., retrieval). However, instead of maintaining the cataloged software itself, GAMS provides transparent access to multiple repositories managed by others. We use the term "virtual repository" to describe such a system.

The GAMS project started in the early 1980s as an effort to catalog software supported for use on NIST's central computing facility (several printed catalogs were produced, the last in 1990). Because of this, GAMS primarily indexes software, both public domain and commercial, which has been installed for use by NIST staff. (Source code of commercial software is not redistributed by GAMS, although items such as documentation and examples are often available thanks to the support of vendors, such as Visual Numerics, NAG, and Cray Research.) More recently the scope of the system has been broadened to include external repositories of publically available software, although only one, *netlib*, has yet been indexed. Also, NIST now supports public access to GAMS to demonstrate the possibilities afforded by community resources of this type. This service was officially announced to the public in the spring of 1994, and there have been some 3000 users each month since then. (See sidebar for information about using GAMS.)

An important feature of GAMS is the taxonomy of mathematical and statistical problems it uses to classify software. The GAMS problem classification system [2] [3] is a 736-node tree-structured taxonomy extending to seven levels (Figure 1). This taxonomy has been adopted for use by many library developers and scientific computing centers (see, for example, [5] [7] [6]). Each software module within GAMS is assigned one or more problem classifications. The taxonomy can then be used as a rudimentary decision tree to help focus in on a group of software modules for a particular problem. We also maintain a thesaurus of mathematical terminology. These terms are mapped to the problem taxonomy, making it possible to translate alternative terminology into that used by the taxonomy. This provides the basis for keyword searching in GAMS.

Another useful concept supported by GAMS is that of "filters." As expected for a system that classifies more than 9000 problem- solving software modules by a 736-node taxonomy, many classes are populated by a large number of modules; hence, even plowing through the modules in a single class can be a chore. With the GAMS filters, the user browsing a particular problem class specifies preferences in a number of areas: language (e.g., Fortran, C), precision (e.g., single, double, multiple), access (e.g., free, proprietary), package, and repository. GAMS presents the user only with modules that satisfy the current filters.

Architecture

The GAMS system is based on a client-server architecture (see Figure 2). The GAMS server runs continuously at NIST, listening for TCP/IP-based socket connections from remote clients. When a client connects to the server, it uses a simple ASCII protocol (gamsp) to request and receive information from the server. The interactions are at a fairly high level, corresponding to queries like the following: What classes are related to the keyword "regression"? What are the subclasses of class D2? What modules are classified at class D2a1? What are the retrievable components of module 7754? Send Source of module 7754. Recently, some 30,000 transactions of this type are being processed each month.

The server has a small database that contains such items as module and package names, brief descriptions, classifications, and other data. In addition, it knows what retrievable components are associated with each module and package and

How to Use GAMS

Local clients. The GAMS Unix clients (*gams* and *xgams*) can be obtained by anonymous FTP from enh.nist.gov (129.6.16.1) in the directory [GAMS].

Guest login. Guest logins are available for those who wish to try out the GAMS clients. To do this, telnet to gams.nist.gov (129.6.80.11) and log in as user gams (for the command line interface) or xgams (for the X Window System interface).

Gopher. A gopher gateway to the GAMS guest logins is provided on gams.nist.gov.

World Wide Web. GAMS can be accessed by World Wide Web browsers like Mosaic or Lynx by specifying the URL http://gams.nist.gov/. Mosaic can be obtained by anonymous FTP from ftp.ncsa.uiuc.edu.

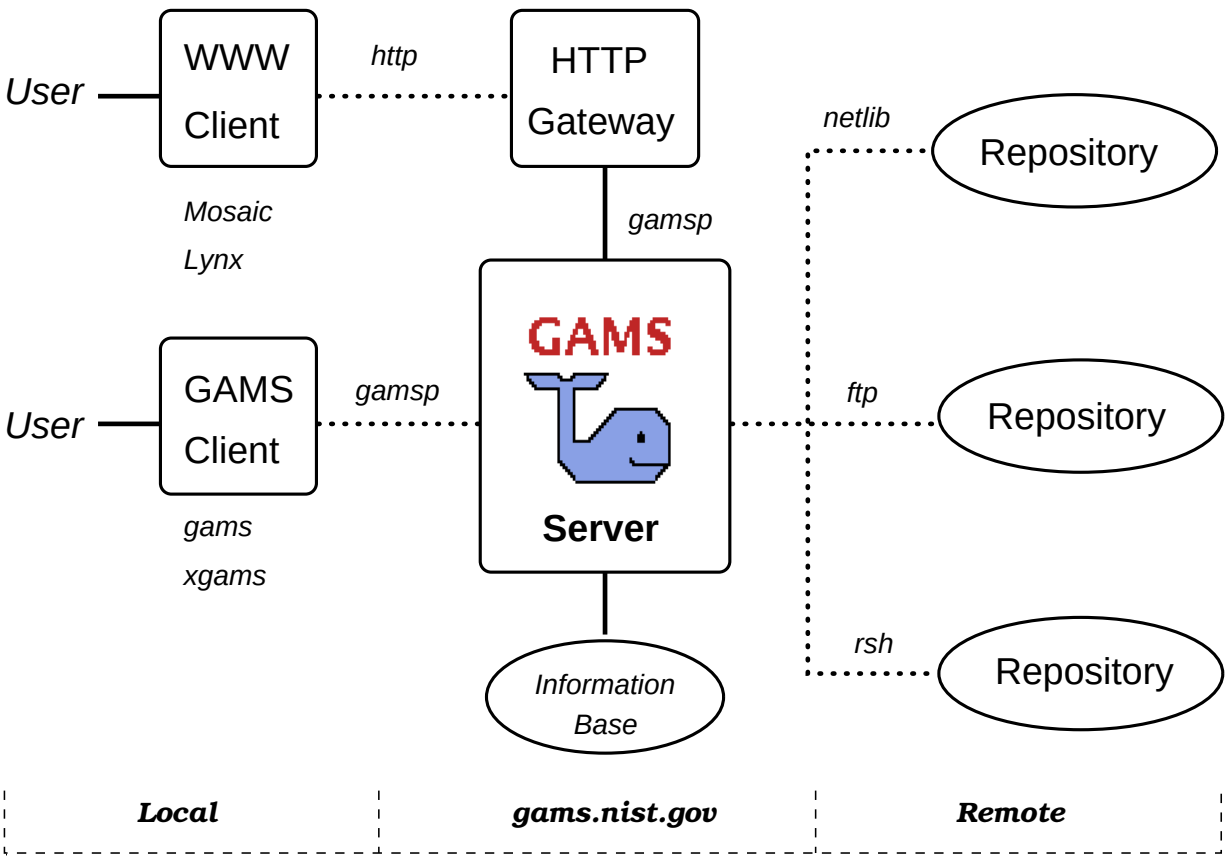
Figure 1: Topmost level of the taxonomy, with problem F expanded.

A. Arithmetic, Error Analysis
B. Number Theory
C. Elementary and Special Functions
D. Linear Algebra
E. Interpolation

F. Solution of Nonlinear Equations
F1. Single equation
F1a. Polynomial
F1a1. Real coefficients
F1a2. Complex coefficients
F1b. Nonpolynomial
F2. System of equations
F3. Service routines

G. Optimization
H. Differentiation, Integration
I. Differential and Integral Equations
J. Integral Transforms
K. Approximation
L. Statistics, probability
M. Simulation, stochastic modeling
N. Data handling
O. Symbolic computation
P. Computational geometry
Q. Graphics
R. Service routines
S. Software development tools
Z. Other

Figure 2: The architecture of the GAMS system.



how to retrieve them from external repositories. When asked by a client to retrieve a component, the GAMS server connects to the external repository in real time (i.e., it too becomes a client), downloads the requested object, and forwards it to the GAMS client.

The GAMS clients manage the interface with the user, connecting to the GAMS server to obtain data as needed. Several such clients, including a simple command-line client (*gams*) and a Motif-based graphical user interface for the X Window System (*xgams*), have been developed for use on Unix systems. A World Wide Web gateway to the GAMS server was developed recently, which means that popular Web browsers like Mosaic and Lynx can also be used to access GAMS. This provides a means of access for PC and Macintosh users.

Looking Ahead

The next step for systems like GAMS is to provide expert- level advice on software selection, and within the context of GAMS we are concentrating on more intelligent schemes for selecting software in a given class. To do so, we are focusing on extensions to the filtering mechanism that support both simple knowledge representations and effective user interactions [1].

We are also beginning work on a substantial revision to the GAMS problem classification system. Much of the current taxonomy is more than 10 years old, and the emergence of software for new problems, accumulated experience with the system, and our expert advisory system goals all provide compelling reasons to start anew. We are very interested in having experts in mathematical and statistical software in a variety of subdomains involved in this project. Please contact us if you are interested!

Although the evolution will probably be much slower than suggested by recent media hype, we believe that experiences in the research sector with systems like GAMS demonstrate that the information superhighway indeed provides a promising means for disseminating useful information that can increase the productivity of its users.

References

[1] R. F. Boisvert. The architecture of an intelligent virtual mathematical software repository system. *Math. & Comp. in Simul.*, 1994. to appear.

[2] R. F. Boisvert, S. E. Howe, and D. K. Kahaner. GAMS—a framework for the management of scientific software. *ACM Trans. Math. Softw.*, 11:313–355, 1985.

[3] R. F. Boisvert, S. E. Howe, and D. K. Kahaner. The Guide to Available Mathematical Software problem classification system. *Comm. Stat. – Simul. Comp.*, 20(4):811–842, 1991.

[4] J. J. Dongarra and E. Grosse. Distribution of mathematical software via electronic mail. *Comm. ACM*, 30:403–407, 1987.

[5] eLib is the electronic library of the Konrad-Zuse-Zentrum fuer Informationstechnik Berlin (ZIB). To use eLib, telnet to elib.zib-berlin.de (130.73.108.11) and login as user elib. eLib can also be reached via the ZIB Web page at URL http://elib.zib-berlin.de/.

[6] The Numerical Algorithms Group, Ltd., Oxford, U.K. *The NAG Fortran*

Library Manual, June 1991. Mark 15.

[7] Visual Numerics Inc., Houston, TX. *IMSL MATH/LIBRARY User's Manual, FORTRAN Subroutines for Mathematical Applications*, September 1991. Version 2.0.

Ronald F. Boisvert is leader of the Mathematical Software Group at the National Institute of Standards and Technology, Gaithersburg, MD 20899. You can find out more on the Web using the URL `http://gams.nist.gov/RBoisvert.html`. *He can be reached by e-mail at boisvert@nist.gov.*

Disclaimer: Commercial products are identified in this article in order to facilitate understanding. Such identification does not imply recommendation or endorsement by NIST, nor does it imply that the products identified are necessarily the best available for the purpose.

This article appeared in the *SIAM News*, Volume 27, Number 8, October 1994, pp. 1,8.