Mathematical and Computational Sciences Division

NISTIR 6297

Information Technology Laboratory

Preprint

Design of a Digital Mathematical Library for Science, Technology and Education

Daniel W. Lozier, Bruce R. Miller and Bonita V. Saunders

February 1999

U. S. DEPARTMENT OF COMMERCE
Technology Administration
National Institute of Standards and Technology
Gaithersburg, MD 20899

NIST

PREPRINT

This paper has been accepted for presentation at the IEEE Advances in Digital Libraries Conference, Baltimore, Maryland, May 19–21, 1999. It will appear in final form in the conference proceedings to be published by the IEEE Computer Society.

Design of a Digital Mathematical Library for Science, Technology and Education *

Daniel W. Lozier, Bruce R. Miller and Bonita V. Saunders
National Institute of Standards and Technology
100 Bureau Drive, Gaithersburg, MD 20899-8910
dlozier@nist.gov, bruce.miller@nist.gov, bonita.saunders@nist.gov

Abstract

The concept of a digital library is of proven worth because of its ability to provide dramatic capabilities that are impossible with traditional print media. We are interested in providing such capabilities for scientific, technical and educational users of mathematical reference data. Our attention is focused on the highly specialized field of mathematics that is concerned with the properties, application and computation of the elementary and higher mathematical functions. Calling upon domain experts worldwide for assistance, the National Institute of Standards and Technology is conducting an ambitious project to construct, ab initio, a comprehensive and authoritative Web resource on this subject. The need to make effective use of the latest developments in digital library research is a major focus, as is the development of content. In this paper we discuss our approach to such difficulties as the representation, display and manipulation of symbolic expressions, numerical data and graphical visualizations, and we describe a prototype Web site that has been constructed to test, evaluate and advance the NIST Digital Library of Mathematical Functions project.

Keywords: Database, Digital Library, Document Conversion, Mathematics, Special Functions, Visualization, World Wide Web.

1 Introduction

The body of knowledge in the sciences, engineering, and mathematics is vast and increasing at an exponential rate. By reading specialized journals and attending conferences regularly, an individual researcher can keep abreast of developments within a relatively narrow field. But when the need arises for results from quite different fields, a researcher may have to invest a tremendous amount of time immersed in the literature; even then, he or she may not succeed in arriving at, and assessing the reliability of, the pertinent information. Digital library technology, coupled with painstaking development and validation of comprehensive reference data, has the potential to minimize this general problem.

Pure and applied mathematics are the most pervasive disciplines in science and engineering. Mathematical definition is the key to uniform and accurate utilization of technical knowledge. Up-to-date refinements of fundamental mathematical techniques, such as approximation of functions, solution of ordinary and partial differential equations, and statistical analysis, provide the underpinning for all modern quantitative science. The increasing reliance of scientists and engineers on mathematical modeling and simulation, the growing use of symbolic and numerical software, and the rapidly developing capabilities of the Internet and World Wide Web, present challenges and opportunities for a comprehensive standardization of mathematical knowledge which supports new levels of multidisciplinary communication.

The authors are part of a NIST team effort to collect, organize, validate, develop, and disseminate a comprehensive and evolving digital library pertaining to mathematical functions. This library is being called the *Digital Library of Mathematical Functions*, or DLMF. The reason for beginning with this particular branch of applied mathematics, instead of,

*This work is supported in part by the Defense Advanced Research Projects Agency under contract DAAH 04-95-1-0595.

Certain commercial software products are identified in this paper. In no case does such identification imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the products are among the best available for the purposes they serve.

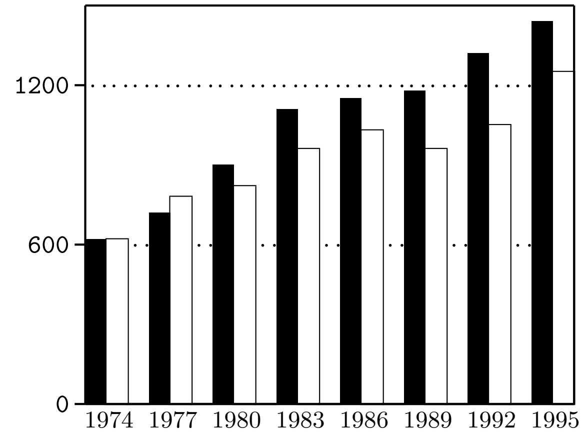

Figure 1. Selected yearly citations to AMS 55 (black bars) compared with total citations scaled by 0.00147 (white bars). Data from Science Citation Index.

say, numerical analysis, is NIST's direct experience with the 1964 *Handbook of Mathematical Functions* [1], known also as AMS 55 (for Applied Mathematics Series No. 55). This work has had unique influence among individuals who apply mathematics to the solution of real-world problems, e.g. engineers, physical scientists, and statisticians. Such users have come to regard AMS 55 as the definitive source of reference information on the "special functions" of mathematics.

AMS 55 is one of *the most frequently cited works in the scientific literature*. Even though it is 40 years out of date (never having been revised), the number of citations to it continues to rise annually, not only in absolute numbers *but also as a fraction of the total number of citations made in the sciences and engineering each year*; see Figure 1. Currently, about once every $1\frac{1}{2}$ hours of each working day some author, somewhere, makes sufficient use of this handbook to list it as a reference. Moreover, the journals in which these references appear range widely over the sciences and engineering; see Table 1.

The target date for completion of the public version of the DLMF, which will be freely accessible from a Web site at NIST, is late in 2002. A distinctive characteristic, in comparison with other initiatives in digital library research and development, is the emphasis on original development of detailed and authoritative content; see § 2 of this paper. An equally important thrust is the dissemination of mathematical reference data as a digital library on the Web with provisions for state-of-the-art indexing, searching, navigation, cross-referencing, linking, downloading, and so on. A prototype Web site with a newly written chapter on Airy functions [7] is described, and some of the issues involved in its construction are discussed in § 3. Advanced interactive graphics in two and three dimensions are a valuable aid to qualitative understanding of the properties of mathematical functions; § 4 discusses issues associated with this topic. § 5 discusses the issue of numerical and symbolic computation, including the location and downloading of software; many users will want easy-to-use support in these matters. Application and learning modules are the subject of § 6. These are auxiliary units tailored to the needs of fields outside mathematics itself, with links to the DLMF Web site. This paper concludes with a few final remarks in § 7.

Table 1. Journals with highest number of citations to AMS 55 in the 10-year period 1988–1997. Data from Science Citation Index.

Cit.	*Journal*
498	Phys. Rev. B: Condensed Matter Physics
462	Phys. Rev. A: Atomic, Molecular, Optical Phys.
381	Journal of Chemical Physics
262	J. Phys. A: Mathematical and General Physics
240	Phys. Rev. E: Statist. Phys., Plasmas, & Fluids
231	Journal of the Acoustical Society of America
205	Journal of Fluid Mechanics
183	Astrophysical Journal
182	Phys. Rev. D: Elementary Particles
153	J. Phys. B: Atomic, Molecular, & Optical Phys.

2 Content Development

The first step in content development is to establish that a genuine need exists for a new initiative in special functions.

The citation record shows (see Figure 1) that special functions are important in science and engineering, and that AMS 55 is meeting this need. However, its writing was essentially finished by 1960, and in the intervening decades numerous advances in mathematical knowledge have been made:

- New functions have entered the realm of practical importance. One example among many is Carlson's elliptic integrals which provide a unification of a branch of special functions that had been unchanged since its inception in the first half of the 19th century.

- New fields of application have emerged. For example, classical special functions are used in soliton theory and nonlinear dynamics.

- Analytical developments have occurred, e.g. in asymptotics and nonlinear sequence transformations.
- New properties, including integral representations, integrals, addition formulas and generating functions, have been discovered.
- Numerical developments, e.g. interval analysis, Padé approximations, and boundary-value methods, have come into being.
- Applications of computer algebra and symbolics have come into wide use.
- An enormous increase in computing power has rendered obsolete many standard numerical processes of the 1950's, such as table-making and interpolation, while at the same time increasing the value of others, e.g. integration of defining differential equations in the complex plane.
- Comprehensive software packages have been constructed, both commercial and non-commercial, for generating special functions by sound numerical procedures.
- The dissemination of information has been revolutionized consequent upon the introduction of electronic publishing and computer networks.

All of these developments are important, and all need to be accounted for in a modern compendium on special functions.

Resources other than AMS 55 exist. How do they relate to the DLMF project? Two handbooks of great importance that are contemporaneous with AMS 55, and therefore just as out-of-date, are [4, 5]. More recent handbooks, such as [8, 9, 10], are much less comprehensive and do not meet the overall need identified for the DLMF, which is to provide a thorough and authoritative treatment of special functions as they apply within and outside mathematics. In recent handbooks, a strong emphasis is often placed on computation, in some cases using numerical and programming methods of lesser overall quality compared with standard procedures in the better-known software libraries. Another kind of resource is represented by current software packages, such as MACSYMA, MAPLE, MATHEMATICA and MATLAB, that include substantial support for special functions as well as broad support of other useful mathematical fields. The fact that companies expend funds in this way for special functions is another indicator of their perceived importance. However, it must be stressed that *these systems do not provide a substitute for an authoritative reference compendium.* In fact, AMS 55 continues to serve as a source of reference information for users of these systems, as well as for the system developers; the DLMF will continue in serving this purpose.

The second step in content development is content definition. Which topics should be included, and which excluded?

Our approach here is to call upon recognized domain experts. An invitational workshop held at NIST in the summer of 1997 resulted in a tentative list of 34 chapter headings. The majority treat individual function classes, such as the elementary functions (exponential, hyperbolic and trigonometric functions, and their inverses) and higher functions (Airy functions, Bessel functions, Legendre functions, and so on); the remainder deal with closely associated topics such as algebraic, analytical and numerical methods. Further refinement of the exact content to be included will result from regular meetings, augmented by email and telephone communications, of an editorial board made up of the four NIST editors and ten prestigious associate editors from the U.S. and abroad.

The third step in content development is the actual content generation, together with quality control. For the DLMF this will consist of contracting with highly qualified authors who will be selected by the NIST editors. The associate editors have all committed themselves to provide substantial assistance to the NIST editors in this and all other important editorial decisions in their respective subfields of expertise. They will also assist in reviewing the written material submitted by the authors. After the content is put into the DLMF Web site by NIST staff, the original authors will be asked to review it, and further review will be arranged by contracts with qualified validators who are independent of the original authors.

The entire project requires substantial funding, and its completion date depends on the funding rate. External funding is being sought to augment committed internal NIST funding. Under the best possible funding scenario, completion of the DLMF project will occur in 2002. The DLMF will require a continuing low level of funding indefinitely. NIST is committed to continuing maintenance under the auspices of its congressionally mandated Standard Reference Data Program.

3 Web Site

In this section, we address some of our design goals for the DLMF Web site, and the implications these goals have for the information architecture of the project. We then give an overview of how we plan to implement that architecture, particularly, how we plan to translate the material our authors will provide into an appropriate internal representation which satisfies the needs of the site.

The bulk of this section describes the plans for the ultimate implementation of the DLMF. However, we have experimented with many of the features described here in a "mock up" of the eventual site (See http://dlmf.nist.gov/, particularly Chapter 11 on Airy Functions[7].), which was prototyped using a highly modified version of LaTeX2HTML.

3.1 Design Goals and their Implications

Let us begin by considering a sampling of the kinds of operations we wish to provide based on our experiences with AMS 55 and its users.

1. Browse through information on a particular special function.
2. Search for a particular kind of formula or information relating to a given property of a particular special function.
3. Find the solution of a particular differential equation, or find the closed form of a particular series expansion.
4. Obtain detailed information about the history, sources, derivation or bibliographic references for a particular formula or section.
5. Cut&paste a piece of mathematical, tabular or other text from the DLMF into an arbitrary user application in LaTeX, Fortran, Mathematica, Postscript, or other format.
6. Print a quick reference guide of the main properties of a special function, or of definitions of all special functions, or of *any* other specific property of a user-specified selection of special functions.

The first three of these user operations primarily concern how a user *navigates* the site, going from the top of the site to some particular piece of information within it. Such traditional tools as indices and a table of contents are as important to this task as they ever were — although they are not as often found on the Web as they should be. A clearly delineated hierarchy with non-overlapping topics at each level is essential for generating a table of contents that a user can quickly and confidently navigate; this requirement is too often ignored. In addition, extensive *meta-information* indicating properties not only of sectional units, but also of formulas and graphics, is needed to construct the several multi-level indices that users might need. This latter information is also essential in constructing automated search tools.

Meta-information attached to each unit or formula can provide much more than indexing capabilities. It provides a natural mechanism to attach information about historical developments, derivations and so forth. This is information that would normally be hidden from the user. But, when a user wishes to go deeper into a subject, that information can be collected and presented, say, by following a hypertext link associated with each formula. Thus, augmenting the text with abundant meta-information helps satisfy items 4 and 5 on our list. See Figure 2 for an example of the meta-information associated with a formula in the mockup site.

To be able to construct quick reference guides (item 6) which collect a variety of different objects (e.g. sections or formulas) satisfying given criteria suggests a slightly different requirement. Attributes of sections and formulas, defined in the meta-information or implied by the sectioning, must be usable in determining the *role* of the object in its context. In this way, we can automatically find, for example, *definitions* of all *functions* in the DLMF; from these collected objects we construct the desired virtual document. Broadly speaking, these attributes define different views of, or slices through, the document. Consequently, in addition to assigning the appropriate attributes to all units and formulas, we must also take care that they are written in such a way as to be useful outside their original context.

Taken together, these implications require us to carefully lay out the overt, hierarchical organization, that is the default book-like arrangement, that DLMF authors will need to adhere to. They also require an internal representation based more on the semantics of the material than merely the presentation of it; that a unit is a subsection is less important than that the unit deals with a particular mathematical property of a function. They also require us to develop several appropriate vocabularies for classifying the entities as to their roles (e.g. definitions, addition theorems, series expansions, . . .), thier origination (e.g. references, derived from another formula, restrictions on applicability, . . .) and so on.

3.2 Internal Document Format: XML

Given these considerations, we find that an internal document format based on XML (eXtensible Markup Language: http://www.w3c.org/XML/), com-

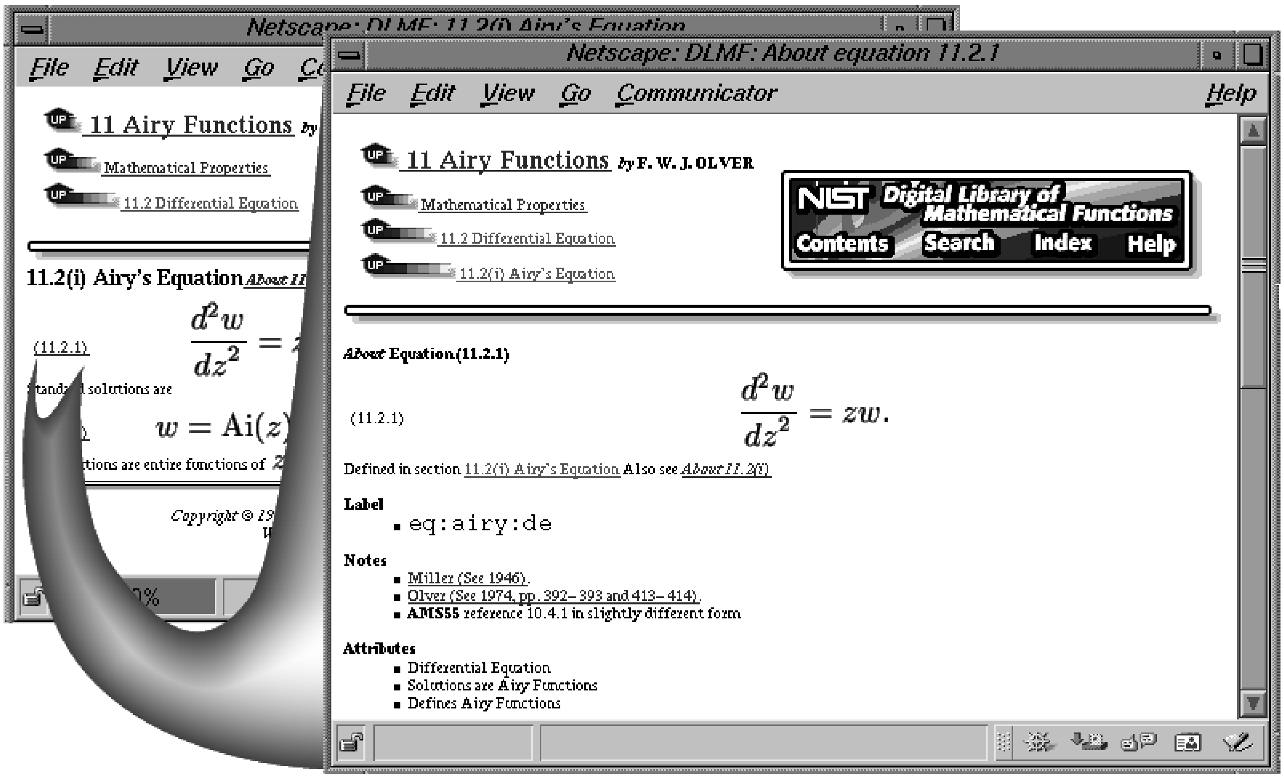

Figure 2. An example of a Meta-information page.

plemented by XSL (eXtensible Style Language: `http://www.w3c.org/Style/XSL/`), will be the most useful and flexible solution. The combination gives a direct method of presentation in browsers that support it. It also supports translation to other needed formats, such as: HTMLwith MathML (Math Markup Language: `http://www.w3c.org/Math/`), LaTeX and PostScript.

Additionally, we will employ the XML application OpenMath (`http://www.openmath.org/`) (or the content model of MathML) to represent the mathematical formulas themselves. OpenMath allows us to represent the semantics of the mathematical formula — what it means and not just how it looks. This aspect is essential to enable the use of these formulas in any other way besides looking at them. Further, OpenMath is being developed as an interchange format for mathematical information and reader modules are expected for many different applications such as computer algebra, text formatting and graphics systems.

3.2.1 Document Structure We do not, in this report, describe a complete DTD (Document Type Definition. See the XML site for more information.). We do, however, give an overview of the main features of the anticipated document structure. In Table 2, we give a schematic of the structure of a chapter on a particular special function (there will be other types of chapter in the DLMF, with different structural requirements). Sections at the lowest level shown will typically include nested subsections and paragraphs, as well as formulas and graphical objects.

3.2.2 Meta-Information Attributes Every structural unit described above, including formulas and graphical objects, can and should be annotated with a rich set of attributes. The following list describes some of the main classes of annotation that we are considering.

attribute: Names of properties satisfied by this unit.

citation: Citation of original information sources.

note: Additional commentary describing the unit.

The following would apply only to formulas.

derived_from: Derivation of one formula from another.

applicability: Indicates that the formula is valid only when arguments or variables are restricted to the specified ranges.

Table 2. Section headings for a special function chapter.

Special Function
- Introduction
  - Overview
  - Notation
  - Definition
  - Graphs and Visualizations
- Mathematical Properties
  - Differential Equations
  - Integral Representations
  - Relation to Other Functions
  - ⋮
- Computations
  - Methods of Computation
  - Tables
  - Approximations
  - Software
  - ⋮
- References

3.2.3 Database of Sectional Units Finally, each structural unit (including graphics and formulas) will be entered into a database, indexed (at least) by the annotations described above. By this means, collections of such document fragments matching various search criteria can be found. Reassembling these pieces allows the construction of a wide variety of quick reference guides or other distillations of material, whether we had planned them in advance or not.

3.3 Creation of XML Documents: Translation from LaTeX

There is one last complication to this grand plan, which is the question of how to create the XML documents as described above. The chapters of the DLMF will be written by various experts around the world. XML is rather new, and unlikely to be familiar to these authors. In contrast, LaTeX is the *lingua franca* for exchange of documents in the mathematical community; most mathematical authors are familiar with it, and it generates beautifully typeset output. While the markup of LaTeX is more semantically oriented than that of raw TeX, it is still primarily oriented towards presentation, and this introduces ambiguities; see §3.3.2, below.

It is conceivable that we might manually translate the LaTeX documents into the desired XML format. However, we may anticipate a period of editing and enhancing the information of each chapter. Since the author may not be prepared to deal with our XML document, we would exchange corrected versions in LaTeX format. Thus a document may have to be repeatedly converted from LaTeX to XML; indeed, many subtle problems with a given document (missing meta-information, ambiguities, etc) may only be revealed by post-processing of the XML! Consequently, we require this translation to be highly automated; if user intervention is required, we should at least be able to augment the source with declarations which would automatically resolve those ambiguities on subsequent processing.

3.3.1 Style files For the most part, the issues involving document structure and meta-information will be easily handled by the definition of a LaTeX *style file* for use by the authors. Macros defining our view of the document structure are easily defined, allowing authors to process, print and proofread their drafts in the usual way. On the other hand, these macros will be recognized by our translation system and used to construct the desired XML representation of the material.

Similarly, a set of macros such as

```
\attribute{addition_theorem}
```

are used to assign the various meta-information to the containing sectional unit or formula. When processed by LaTeX, these generate indices or margin notes; they can even be ignored if they are not being proofread. When processed by our translation system, however, they will be embedded in the XML output, and also extracted to construct the required indices and database attributes.

3.3.2 Mathematical Ambiguities The ambiguities of LaTeX are most apparent in the case of markup of mathematical formulas. As two simple examples, consider the expression `f(x+y)` (which would display as $f(x+y)$), and `\frac{df}{dx}` (which would display as $\frac{df}{dx}$). In the first case, the juxtaposition of `f` and `(x+y)` might represent either function application or multiplication. In the second case, this fraction construct is a common idiom for a derivative, but *might* really be intended to represent the (unsimplified!) fraction. Generally, human readers aware of the context of the formula will immediately recognize the intended interpretation. Computers often fare less well.

It is, of course, here that our translation software will be most challenged. And given the high confidence level we hope to achieve with the DLMF, we are not inclined to rely on heuristic methods of analysis. To this end, we devise what may be more tedious, but hopefully more reliable, methods to disambiguate the formula.

To handle the first kind of problem, we require that every symbol used must be declared to clarify its type as either a variable or function; a scalar, vector or matrix; real or complex; etc. Thus, if f is a function and $x + y$ is not, we interpret `f(x+y)` as a function application; otherwise the expression represents a product. Given these declarations, a minimal type inference system can be expected to be sufficient for resolving these cases.

The second kind of problem is handled simply by requiring the author to say what he means; macros such as `\deriv{f}{x}` are introduced to express explicitly that a derivative is intended. Nevertheless, we still must expend some effort to recognize expressions like `\frac{df}{dx}` in the document as *potentially* standing for a derivative.

Ultimately, we must develop a rather complete set of such macros to disambiguate all the notations and usages that we expect to find in the author's materials. These macros would be included in the provided style files, and authors would be encouraged to use them. We do have the advantage, however, of not having to deal with *all* of mathematics, for which the richness of notations with multiple interpretations becomes quite overwhelming.

4 Graphics and Visualization

A Web-based digital library offers significant advantages over printed media for the presentation of informative graphics. Whereas graphical representations in [1] were sparse and restricted to static 2D plots, in the DLMF, dynamic 3D visualizations of complex special functions will complement 2D and 3D still images. The judicious use of graphics not only will help scientists and other technical users gain a deeper understanding of special functions, it will also assist us in making some parts of the DLMF accessible to educators, students and others who want a short introduction to the field.

The development of effective graphical displays in the DLMF presents several challenges. To insure uniformity throughout the DLMF, we must come to a consensus about the location, frequency, and type of visualizations to be placed in the system; but close collaboration with individual authors will be required to determine the specific needs for each chapter. A reliable means of computing accurate data for the visualizations must be found, and the author must decide which special features of a function, such as zeros, poles, and other singularities, should be emphasized. At the same time, the plotting range and scaling of the function must be determined to illustrate best those features. Another issue is deciding the type of file formats to be used, while also considering the availability of any additional Web plug-ins that would be needed by the user. We address some of these challenges by examining what we have done for the chapter on Airy functions in the prototype Web site.

4.1 Static and Dynamic Visualizations of Special Functions

The author of the chapter on Airy functions[7] specified which Airy functions should be displayed and suggested the ranges for the plots. To obtain reliable data we used a double precision Fortran routine for the calculation of Airy functions written by D.E. Amos [2]. In general, all of the authors of the DLMF should be knowledgeable about the latest computational techniques being used for the functions in their chapter and should therefore be able to point us to the best means for obtaining reliable data.

For both the still images and dynamic visualizations we began by using available packages such as Matlab and Mathematica to plot the data so that we could examine the graphical representation and adjust the scaling to bring out interesting features. Also, considerable effort was spent determining the most informative views for the still images. The still images were stored in GIF or PostScript format. To obtain dynamic visualizations, we wrote a C program to convert the data to VRML (Virtual Reality Modeling Language; `http://www.vrml.org/`) format. VRML is a standard 3D file format for describing the geometry and movement of a 3D virtual world. We chose VRML because of its accessibility on the Web and its interactive capabilities. Also, VRML browsers for a variety of platforms can be freely downloaded. However, it is not a foregone conclusion that the completed version of the DLMF will use VRML. We have to address such issues as whether VRML browsers will continue to be readily available and what to do if a browser is not available for a specific platform. We will also examine the feasibility of using alternatives to VRML such as Java 3D which would not require the user to download a browser. Currently the user is given the option of viewing a still 3D image if a VRML browser is not available. Figure 3 shows a VRML display of the real part of Airy function $\mathrm{Ai}(z)$. The browser controls allow the user to rotate the figure, zoom in and out, and move the figure in an arbitrary direction.

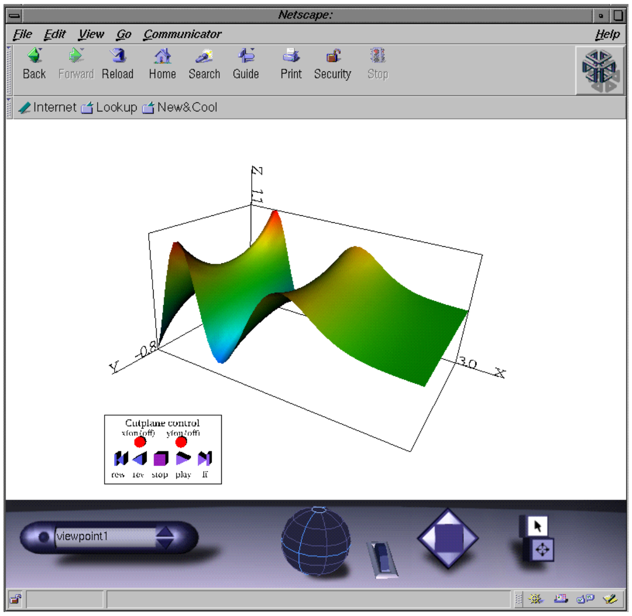

Figure 3. VRML **display on CosmoPlayer.**

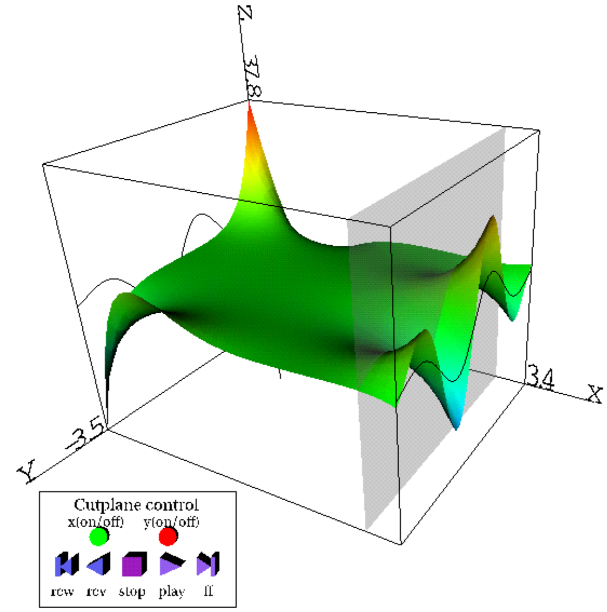

Figure 4. VRML **display with X direction cutting plane.**

4.2 Intersection of 3D Surfaces with Cutting Planes

In addition to the standard controls that come with the VRML browser, Figure 3 shows a panel labeled "Cutplane control" which gives the user additional capabilities. We used VRML to create files that would generate cutting planes through a 3D surface. By manipulating the panel controls, a user can study the change in the intersection as a plane is moved through a surface. Currently, the cutting planes are limited to planes perpendicular to the X and Y coordinate axes, but we are working on an extension to the Z direction. Future work will extend the capability to an arbitrary direction.

When the user clicks the X button on the Cutplane control panel, a bounding box appears around the figure along with a cutting plane that moves perpendicular to the X axis. The user moves the plane by clicking on the second row of buttons which operate like those of a VCR. The plane moves in sync with the projected intersection curve, displayed on opposite faces of the bounding box as shown in Figure 4. The controls operate similarly in the Y direction. We are working on the addition of a slider bar to the control panel to give the user more flexibility and make it easier to stop the plane at desired points.

To implement the cut plane control, we used VRML reusable components called PROTO's. Our Cutplane PROTO displays the plane and searches the surface data to determine which points are closest to the specified plane. Linear interpolation is then used to obtain the coordinates for the intersection. All the surfaces we have done to date intersect the X or Y planes in continuous curves. If the surface contains holes, then the intersection curves will be disconnected at some locations, so we have to be careful about how we connect the points. Such a situation is the norm for the Z direction. The intersection of the Z direction plane with the surface is the contour curve for that level which, in general, is not a single continuous curve. Therefore, we are testing various packages for contour plotting to use in determining the Z direction intersection points. In testing the effectiveness of the packages on different machines we are finding that an acceptable speed for the VCR controls on one machine may actually be too fast on another. For that reason, we may decide that the slider bar, which gives the user more control, is a better choice.

4.3 3D Clipping

An unexpected issue arose when some plots were rescaled to emphasize interesting features. We found that most of the packages we used clipped the surfaces in an unsatisfactory manner. Some packages simply reset values above a certain height to the same constant, producing the shelf effect illustrated in the plot of Airy function $|\mathrm{Bi}'(z)|$ shown in the first graph in Figure 5

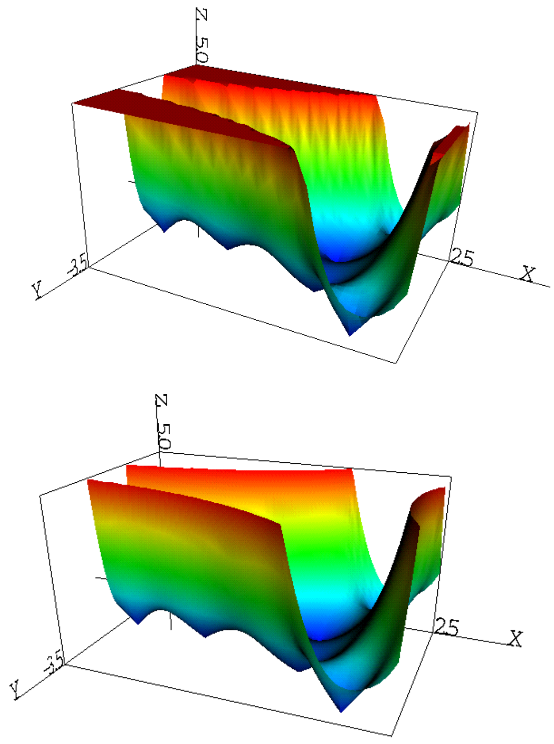

Figure 5. Unclipped and clipped graphs.

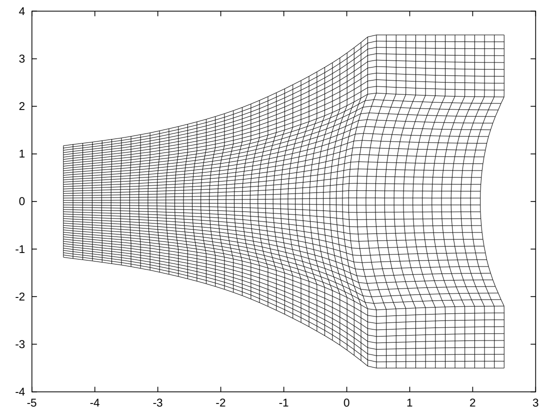

Figure 6. Contour mesh.

and also seen in Thompson [9]. Others suppress the plotting of points where the function value is greater than a specified number, but this may produce plots with jagged edges that are equally misleading.

Another problem is the extraction of the clipped data and its translation to a format we can use. In at least one package, we discovered that although the clipped surface looked fine on the screen, the output of the plotted data included the entire surface instead of the clipped surface.

We have been unable to find a package that meets all of our needs. Therefore, we are doing some work in developing our own techniques. We created the smoothly clipped second graph in Figure 5 by using techniques from the field of mesh generation. First, we selected the height at which we wanted to clip the function, $Z = 5$. We then used the $Z = 5$ contour curve of the function to construct a boundary for our domain. A boundary fitted mesh was placed on the domain as shown in Figure 6. By computing the Airy function only at values on the mesh we obtained the smoothly clipped surface plot in the figure. We discovered that clipping the figure by this technique also smooths the shading, which is based on the height of the function at the grid points. This is probably because the mesh lines are close to being contour curves.

Creating a boundary-fitted mesh based on contour information about the function is an ideal solution for many graphs, but it is clear that the mesh generation problem can become quite complicated for more complex special functions that have features such as steep gradients, zeros, or poles. For example, a contour mesh for the gamma function would be multiply-connected with several holes. We may want to look at triangulation techniques to handle more complicated domains, although doing so may also affect the way we write the cutting plane software.

5 Numerical and Symbolic Computation

The chapter on Airy functions in the prototype Web site includes four subsections on the general subject of Computations; see Table 2. The first, Methods of Computation, gives a list of general approaches that have been used to construct algorithms, with references to the literature. The approaches identified in this subsection are distinguished by their generality, i.e. in principle they can be combined to achieve any degree of precision because they start from analytical definitions of the functions: series expansions, differential equations, integral representations, and representations in terms of other functions. Numerical considerations such as convergence, accuracy and stability are mentioned briefly. The second subsection, Tables, gives references to published numerical tabulations. Such tables are of occasional use in validating mathematical software but almost always their suitability for this purpose is severely limited by their inadequate precision and range in comparison to the capabilities of current software. Because of the existence of numerical software, tables are rarely used today for their original purpose of providing function values for pencil-and-paper calculations by interpolation between the tabulated entries. Therefore, the DLMF will not include voluminous static tables, which occupied over one-half the pages of AMS 55.

The third Computations subsection is Approximations. Here, references are given to papers that provide fixed finite-precision approximations, usually for restricted ranges of the independent variables. These are valuable when, as is often the case, their execution speed is fast in comparison to other methods. They are often found at the heart of numerical subroutines in software libraries. The fourth subsection, Software, is the one likely to be the most often consulted by DLMF users. Here distinctions are made among *programs* that have been constructed and published by an original author; *libraries* that have been produced by gathering programs and imposing uniform conventions with respect to documentation, style and handling of errors; and *systems* that provide an interactive command-line interface. The subsection provides a classification and listing of published and commercially available software, complete with the pertinent restrictions on the ranges of the independent variables and the precision of the computed results. It also provides immediate access to documentation, and even to source code, via links to GAMS (`http://gams.nist.gov/`) (the Guide to Available Mathematical Software) and Netlib (`http://www.netlib.org/`); see also [3].

Eventually the DLMF will include a more interactive facility for computing numerical values of special functions. This will allow a user to specify precision and the ranges of independent variables quite arbitrarily. The actual computation may take place on dedicated computers at NIST or, alternatively, in JAVA code downloaded to the user's local environment. A number of research problems remain to be solved before such a facility can be put fully into place, even for a small selected subset of mathematical functions. The most important of these problems is the requisite error analysis, which is very demanding analytically, and which must be put into a computable form. This is essential to be able to assure that the computed results are accurate to the precision specified. Nevertheless, a prototype facility for selected functions is under construction that will apply within certain limits of precision. It will have a strong likelihood (but not a guarantee) that the precision criterion has been met.

One useful purpose for such a facility is a "software test service for special functions;" see [6]. Another is to support user-driven visualizations of mathematical functions in which ranges of independent variables are specified by the DLMF user.

The role of symbolic computation in the DLMF is still being discussed. One possible role is to provide a way to determine mathematical equivalence of expressions, for example when a user is searching for a mathematical formula which he or she expresses in a form that is different from but equivalent to a formula that is encoded in the DLMF database.

6 Application and Educational Modules

The DLMF is envisioned not only as a basic resource for scientific professionals but also as a foundation for innovative, discipline-specific "application modules" that can, for example, eliminate some of the vexing variations that occur in the use of mathematical functions in different application areas. Some of these variations are merely notational but most are related to the fact that real-world applications involve physical constants, normalization conventions, and special conditions that have no place in a purely mathematical treatment. Since mathematical functions are intrinsic in so many different fields, no attempt can be made to cover all fields within the DLMF project. Thus the DLMF will contain only a very restricted set of illustrative examples that have a connection to an application area in chapters where it is appropriate. However, one of the outstanding benefits of a richly interactive and interlinked Web site is the opportunity it presents to construct associated Web sites that are tailored to specific application areas and discuss them in substantial detail. We expect that the DLMF will serve as the repository of core mathematical information for such Web sites. The DLMF project intends to provide two such application modules, in quantum mechanics and electomagnetic theory, as prototypes for others to consider.

The DLMF provides an opportunity to construct educational modules in exactly the same way, and a prototype in this field is being developed also. It will focus on mathematical functions that are introduced in high school, elaborated in university, and used in virtually all engineering and scientific applications. We expect that it will quickly become popular as an innovative learning resource for well-motivated high-school students, university students and technological professionals. It will provide an on-line tutorial with the ability to (i) search for and retrieve formulas and other information from the DLMF, (ii) generate computational results on demand, (iii) generate user-controlled graphics and visualizations on demand, (iv) check the correctness of exercises completed by the student, and

(v) follow links to related Web sites.

7 Concluding Remarks

It is a truism that our world is becoming increasingly mathematized. This has been going on for at least two centuries, and the pace is accelerating. Special functions is a vital branch of mathematics that has deep and far-reaching impact in science and technology. And it is not only the scientific world that is mathematized. Advanced mathematical techniques are central also in economics, finance, scheduling, communications, risk assessment, ...; the list is very long.

Non-mathematicians who are faced with on-the-job mathematical demands need ready access to reliable mathematical information that ranges across a spectrum from critically evaluated reference data, through scholarly synopses, to introductory tutorials. The eventual DLMF Web site will meet needs across this spectrum, and in a highly usable form. This is a natural application area for a digital library, with its ability to provide scientists, technicians and students with powerful interactive tools.

References

[1] M. Abramowitz and I. A. Stegun, editors. *Handbook of Mathematical Functions with Formulas, Graphs and Mathematical Tables*, volume 55 of *National Bureau of Standards Applied Mathematics Series*. U. S. Government Printing Office, Washington, D. C., 1964.

[2] D. E. Amos. Algorithm 644. A portable package for Bessel functions of a complex argument and nonnegative order. *ACM Trans. Math. Software*, 12:265–273, 1986. for remark see same journal v. 16 (1990), p. 404.

[3] R. Boisvert, S. Browne, J. Dongarra, and E. Grosse. Digital software and data repositories for support of scientific computing. In N. R. Adam, B. K. Bhargava, M. Halem, and Y. Yesha, editors, *Advances in Digital Libraries*. Springer-Verlag, New York, 1995.

[4] A. Erdélyi, W. Magnus, F. Oberhettinger, and F. G. Tricomi. *Higher Transcendental Functions*, volume 1. McGraw-Hill, New York, 1953. Volumes 2 (1953) and 3 (1955) printed by same publisher. All volumes reprinted in 1981 by Krieger Publishing Company, Melbourne, Florida.

[5] E. Jahnke, F. Emde, and F. Lösch. *Tables of Higher Functions*. McGraw-Hill, New York, sixth edition, 1960.

[6] D. W. Lozier. A proposed software test service for special functions. In R. F. Boisvert, editor, *Quality of Numerical Software: Assessment and Enhancement*, pages 167–177. Chapman & Hall, London, 1997.

[7] F. W. J. Olver. Airy Functions, Chapter 11, Digital Library of Mathematical Functions Project. National Institute of Standards and Technology, Gaithersburg, MD 20899-8910, U. S. A., 1998. http://math.nist.gov/DigitalMathLib/.

[8] J. Spanier and K. B. Oldham. *An Atlas of Functions*. Hemisphere Publishing Corporation, Washington, D. C., 1987.

[9] W. J. Thompson. *Atlas for Computing Mathematical Functions: An Illustrated Guide for Practitioners, with Programs in C and Mathematica*. John Wiley & Sons, New York, 1997. An edition with programs in Fortran exists also.

[10] S. Zhang and J. Jin. *Computation of Special Functions*. John Wiley & Sons, New York, 1996.