AIAA

A01-16726

# AIAA 2001-0913
# A New Internet System to Improve Numerical Code Development

Theodore K. Lee and Xiaolin Zhong
University of California, Los Angeles
Los Angeles, CA

**39th Aerospace Sciences Meeting & Exhibit**
January 8–11, 2001 / Reno, NV

# A New Internet System to Improve Numerical Code Development

**Theodore K. Lee*and Xiaolin Zhong†**
University of California, Los Angeles, California 90095

## Abstract

Many research areas now extensively use computer codes to calculate solutions to various situations. The use of computer programs to simulate a certain physical process requires the use of advanced numerical methods which must provide an accurate solution. The numerical methods themselves, and the results that they produce, are currently still being presented through traditional mediums such as journals and conferences. The objective of this work was to develop a new numerical code Internet system that would dramatically improve the way that numerical codes are currently developed, tested, used, and presented. The Advanced Remote Code Operating System (ARCOS) was developed with the Java programming language. The advanced features of ARCOS include the ability for remote users to access and execute existing numerical codes through a Web browser, graphically view results through a Web browser, providing a more efficient and accurate method of exchanging data, improving the evaluation of new numerical methods, and increasing accessibility of the numerical codes to interested parties, especially those who are not highly knowledgeable about the code's particular methods.

## Introduction

The Internet has dramatically changed the speed and accuracy with which information is exchanged and used. While most of the focus on the Internet's capabilities has been in the areas of news media [1] and Web commerce [2], scientific communities are just beginning to see the Internet as a new resource for their research [3]. Many research areas now extensively use computer codes to calculate solutions to various problems. The use of computer programs to simulate physical processes requires the use of advanced numerical methods which must provide an accurate solution. Despite the advances of scientific research involving computers, the numerical methods and the results that they produce are still being presented using traditional mediums such as journals and conferences. Unlike other areas of scientific study, numerical code development has evolved into an individual exercise, where results and methods are presented with no additional input or validation from outside sources. Source codes and numerical data files are not made available to others for independent analysis. This method of scientific study is completely different from other aspects of science which involve work such as the derivation of numerical methods, and performing experiments. Previous methods of study encouraged impartial observers to try and re-derive or re-create the results that were initially obtained. Unfortunately, the use of computers to solve for various processes has not resulted in a better method for either numerical code validation or the communication of ideas due to the lack of an efficient method for independent verification. There are currently no effective means through which researchers investigating similar problems may present their advances in numerical methods and share their computational results. This communication gap only impedes the advancement of computational methods and solutions, when in fact, the use of computers should be used to dramatically improve communication.

Previously, the authors had developed an Internet system that was applied specifically to a hypersonic heating numerical code [4]. This system was able to execute both inviscid flow results over simple three-dimensional geometries, as well as compute the surface heating resulting from a hypersonic flow condition. The previous work, however, was designed to be completely applied to only one specific numerical code. The motivation for this study originates from the need for a new system to easily integrate any existing computer program with the Internet. The area of scientific research that should be embracing the Internet is the use of computational simulations, yet this has not happened. Computer codes are now used to simulate virtually every imaginable physical process that is of interest. Results of these computational simulations can be found in many scientific journals and conference presentations. The problem, however, is that computational code development has become a solitary exercise, where codes, methods, and results are not validated or presented completely. The validation of numerical codes is performed by the developer themselves, and not by independent sources. Results presented in papers are only select cases, and usually do not include the entire process in question. Unfortunately, numerical codes can-

*Graduate Student, theodore@seas.ucla.edu, Student Member AIAA
†Associate Professor, Mechanical and Aerospace Engineering Department, xiaolin@seas.ucla.edu, Associate Fellow AIAA

not be easily reproduced, unlike the derivation of numerical methods or a laboratory experiment. Results that are presented are not reproduced by independent sources, because the developer's codes are not available. Independent sources will not write their own numerical code because the development time is too long, and the methods used are not always clear. The developers will not give their source code away because they have proprietary rights to it. As a result, the development of numerical codes and numerical methods has become isolated and slow moving.

The best solution to this communication gap is to begin taking advantage of the Internet. The Internet can be better utilized in the development of numerical methods and computational codes. The Internet has numerous benefits that can be applied to numerical code development and validation. Where previously, only select results were shown in small figures, using the Internet, entire data files can be easily downloaded and examined. Where previously, domain generation and other initial parameters were not clearly stated, they can now also be displayed and downloaded easily over the Internet. And finally, where previously developers were reluctant to reveal their source code, the Internet will allow outside users indirect access to the executable file in order to remotely execute numerical codes, using their own defined input parameters to generate a solution data file which they then can download and analyze for themselves. An Internet system called the Advanced Remote Code Operating System (ARCOS) has been developed that gives anyone the ability to easily achieve all of the functions above.

ARCOS does not only apply to numerical code development and future validation of codes, but it also serves a purpose for many existing programs that were written in the past, but that are still being used today. There are numerous old computer programs that are still being used, both in industry and education. These programs, however, require the user to be directly in contact with the computer that the program is stored on, such as through a telnet session. By using the Internet, access to these old, but useful, programs can be dramatically increased. When a useful numerical code is accessible to more potential users, it increases its usefulness and value to that particular organization.

The advantages of ARCOS can be described as follows :

- This system does not require any advance knowledge of Internet processes beyond embedding a Java [5] applet onto a Web page, and editing simple text files. Scientists may have been wary of placing their numerical codes on the Internet, due to the development time required to successfully develop a system to do so. This system will eliminate the need for scientists to spend valuable time learning Java or client-server communication parameters. This system also gives the developers complete control over what they make available over the Internet, with the ability to restrict access whenever they wish.

- ARCOS allows researchers to be able to remotely execute their current numerical codes from any computer that is connected to the Internet by using only a Web browser such as Netscape or Internet Explorer. Currently, the only way to access and execute numerical codes remotely is by using a telnet session. However, using a telnet session is restricted to the developer himself, and no other users have access in this manner. By allowing access through a Web browser, any remote user can execute the numerical codes. The numerical codes themselves can be written in any programming language. As long as an executable that is able to run on the developer's host machine is available, ARCOS will work.

- This system allows researchers to download numerical data through the Internet that has been produced by computations that were remotely executed. The output that has been produced from a remote execution is automatically placed on a Web page. This eliminates the need for the developers to learn HyperText Markup Language (HTML) [6], and completely automates the production and updating of Web pages.

- ARCOS is designed so that it can be applied to any computational code that requires only simple inputs such as numbers and text. Most scientific codes require some form of input parameters. This system allows the developer the ability to define the input data files required, and the individual inputs contained within each data file. When remote users are about to execute a code, they will have complete control over the input values.

- ARCOS is able to graphically plot data on the remote user's Web browser, as long as the data is stored using the correct format. Plots for one-dimensional up to three-dimensional cases can be plotted with a variety of methods.

Presently, there is no such software package that allows researchers to easily allow remote access to their numerical codes. In addition, there has not been a large movement towards using the Internet as an effective and efficient means of communication by computational researchers. ARCOS is the first system that is specifically designed to be used with scientific numerical codes and to display scientific results. The use of the Internet for scientific computing is an inevitable extension of current numerical code development practices. ARCOS is a useful tool that provides an easy to use method which allows researchers to begin the new process of Internet scientific computing. The potential applications of ARCOS are as follows:

- providing a new method of numerical code validation,
- providing complete solution data files for analysis by interested parties.
- increasing the knowledge of specific numerical methods and their capabilities,
- educating interested users about numerical methods and the physical processes they are able to compute.

## Java Language

The Java language is quickly becoming an industry standard when dealing with various Internet functions. It has many advantages over traditional programming languages such as C++ and Fortran. Java's biggest advantage may be its portability. A program can be written once in Java using any operating system, compiled once, and subsequently executed on a variety of operating systems, including Windows and UNIX based systems. Another advantage of the Java language is that there are numerous class libraries that the programmer can use. These libraries allow for a quicker and more efficient programming strategy, especially when dealing with multimedia applications involving sound, graphics and window interfaces.

ARCOS is written entirely using the Java language. Because of the portability, developers only need to possess the Java executable files in order to immediately make their codes accessible through the Internet. The system is separated into 2 major components. The first component is called a Java applet [7]. A Java applet is a small program that is embedded within a Web page, much like an image is placed on a Web page. It is automatically executed when a Web browser, such as Internet Explorer, opens that particular Web page. The function of the Java applet is to allow the user to input certain parameters and to display certain results. The second component of the system are the two server programs. These server programs must be running on the developer's host machine. The server programs listen and communicate with the Java applets, and perform all of the data file writing and reading that is required. The server programs are also responsible for executing the codes upon receiving proper instructions from the applet. This type of Internet communication is called client-server communication [8], where the applets are considered the client, and the programs executing on the host machine are considered the server.

Recently, a new addition to the Java language called Java 3D [9] was released for use with applications requiring advanced graphical capabilities. This new feature of the Java language allows the programmer to incorporate three-dimensional scenes in the application. Such uses from an scientific standpoint include viewing the numerical grid, color contours, as well as other plots of desired properties along a defined geometry.

## Previous CFD/Heating Application

Previously, the authors had developed a system using the Internet that was applied specifically to a hypersonic heating numerical code [4]. This system was able to execute both inviscid flow results over simple three-dimensional geometries, as well as compute the surface heating resulting from a hypersonic flow condition. We had developed this system to be used in conjunction with our Computational Fluid Dynamics (CFD) and aerodynamic heating codes. The work on integrating a high order accurate CFD inviscid flow solver with analytic heating models was motivated by the need for accurate prediction of aerodynamic heating around hypersonic vehicles by NASA Dryden Flight Research Center [10-13]. The purpose of this system was to allow us to remotely execute our codes through the Internet with a graphical, user friendly interface.

The basis of that system was the ability to perform a number of different tasks related to CFD and heating results. The first set of tasks was to generate and view a new grid, or to view a grid that already exists within the database. The user could define the size of a new grid and immediately view it on the Web browser using the graphical features that Java provides.

The second set of tasks was to begin a new CFD inviscid flow computation, or to view existing inviscid flow results. To begin a new inviscid flow computation, the user would define the freestream flow properties, the numerical methods, generate a grid, and define the boundary condition order of accuracy. Once these input parameters had been defined, the server automatically began the execution of the CFD numerical code. The aerodynamic properties that could be graphically displayed included pressure, temperature, density, Mach number, speed of sound and velocity.

The third and last set of tasks involved using the inviscid flow solution to compute surface heating over the respective geometries. The users could choose to compute a new heating solution, or view an existing heating solution. Users needed to input the heating parameters that are needed, such as choosing between laminar and turbulent models, and the region of the vehicle that was to be considered. The server automatically executed the heating program, and produced a data file containing the surface heating results.

The basis of that system is what was used in designing ARCOS. The fundamental building blocks and functions are similar, in that most scientific programs require some input parameters, and that they produce a numerical output data file. Where before the system was tailor made specifically for that application, ARCOS is designed to be used with any scientific numerical code, with the added feature of automatically creating and updating Web pages.

## Java Applet/Server Development

The core of ARCOS is the client applets and server programs which are both written with the Java language. There are three Java applets that have been developed, along with two Java server programs.

ARCOS was designed with generality in mind. It is able to execute any numerical code, as long as the inputs are in the form of numbers and text. This system also allows multiple codes to be placed on the Web, and executed. When multiple codes are used in conjunction with the same server, the applet allows the remote users to choose which code they wish to execute. By allowing multiple codes to be placed on the Internet, this will allow numerical code developers to not only use their current programs, but to allow them to add additional programs as well. This system also does not allow the remote users any access to the source files, thereby maintaining the developer's proprietary rights. The only elements of the code that a remote user would be aware of are the input data files and the output data files.

The first Java applet is designed to be used when the developer decides to make a numerical code accessible over the Internet. This applet is used to obtain certain information from the developer. A screen shot of this applet is shown in Figure 1. The developer must input the name of the code, the name of the executable file, the directory location of the executable file, as well as his own name. The developer must also input the names of all the input data files, the names of all the individual inputs within these data files, along with a designation of the type of input (number or character string). Lastly, the developer must enter the names of the output data files, and decide whether he wants to make the output available to be downloaded through a Web page. To assist remote users, the developer can also input detailed descriptions of the code itself, the contents of each data file, and of each individual input. A chart of the this client structure is shown in Figure 2.

The applet takes this information that has been input by the developer, and sends it in an organized manner to the server program which will be running on the host machine. The server program is a very small program which should be running in the background on the same machine that contains the Web pages. It is designed to continuously listen for the applets communication. The server receives the information, organizes it, and writes a unique program data file which will store this information. It is possible for the developer to allow remote access to any number of numerical codes, all through the same Web page. Each numerical code will have its own unique program data file. A chart of the server operations is shown in Figure 3.

The second Java applet is designed to be used by the remote users who wish to access and execute the code. The remote users choose which numerical code they wish to execute from a list of codes that have been placed on the Web by the developer. A screen shot of this applet is shown in Figure 4. Once the remote user has selected which code they wish to use, the applet informs a second server program to send back the data that is contained within this particular code's program data file. The Java applet displays the contents of the program data file. Specifically, it displays the parameters that are required, along with any other helpful information that the developer submitted. The applet will then query the remote users to provide the values for the input parameters that the developers specified. Initial flow conditions, grid sizes and a choice of methods may be some of the questions that the remote user will have to answer. A chart of the remote user's interaction with the Java applet is shown in Figure 5.

Once the remote user has finished inputing all of the needed parameters, the applet sends this information back to the server. The server creates a new directory, writes the necessary input data files, and begins execution of the numerical code. A chart of the server operations is shown in Figure 6. If the developer wishes, the input and output data files are added to a Web site database containing previously computed results. The server will automatically update the Web page that displays the results, and add in the newly computed solution. This database will be accessible from a Web page that is designed specifically to display the data files that are available to be downloaded.

The third Java applet will be used to immediately view the results of the numerical codes in a graphical manner. Through the use of advanced Java 3D graphical capabilities, this applet automatically downloads a selected output data file and displays the results. With each output data file that is created through this system, the name and location of them are stored. Once a remote user wishes to graphically view a particular solution, they simply have to choose the appropriate solution from a list of all the stored data files. If the data file is written in the proper format, then the user is able to graphically view the results. The graphical capabilities involve three types of plots. The user may choose to display the grid, the user may choose to display a line (x-y) plot, or they may choose to display a color contour plot. A screen shot of the graphical applet is shown in Figure 7.

While there are various simple maintenance duties that the developers must perform, they are not much more then editing text files, and deleting old data files. There is no need for any prior knowledge about the Java programming language, or the internal communication structure of the client-server programs. This system allows any interested party situated at any computer with Internet access to execute numerical codes for conditions that they require. The use of a graphical user interface allows people who are not fully familiar with the numerical code to easily input desired initial parameters and quickly obtain results. This greatly reduces the amount of time that researchers would otherwise need to study the requirements of the code and the methods available before executing. As more and more code developers begin placing their numerical codes on the Web, the communication of computational results and numerical methods among the scientific community will become much more efficient.

## Ability for Remote Access

The most important aspect of ARCOS is the ability to allow scientists to remotely execute their numerical codes over the Internet in a secure and easy manner, using their existing executable files. While there may be significant interest among the scientific community about using the Internet more efficiently, there have been large obstacles to overcome. Currently, scientific codes that are remotely executable over the Internet have had to be programmed in an Internet executable language, such as Java [14]. There is, however, an enormous amount of numerical programming that has been written in languages such as Fortran or C++ that are currently still being used. It is impractical for scientists to reprogram these codes into another language such as Java, since that would involve too much development time. In some cases, programs that have been written in the past but are still currently being used, do not have an available source code, thereby making any attempt at reprogramming even more difficult. Because of these obstacles, the use of the Internet has not advanced as quickly as it could.

ARCOS has been designed to execute any numerical code regardless of what programming language was used, as long as there is an executable file, and the inputs are either numbers or character strings. Even the source code of the program is not required. The executable file is linked to the server programs, and the server program communicates with a applet. Remote access to the numerical code can be restricted to the developers themselves, or the developers may wish to allow any remote user the ability to access their code. To restrict access to their code, the developer only has to cease the execution of the server programs when they are not in use. With no server programs running, the Java applets are useless since they are unable to communicate with the server to acquire the needed information.

## Efficient and Accurate Information Exchange

Another important aspect of ARCOS is the fact that it takes advantage of the fastest, most efficient, and most accurate method of communication that is available to computational scientists. The Internet allows communication between scientists and engineers to be almost instantaneous, regardless of the distance between them. An example of highly efficient communication can be seen in the e-mail that is used for personal communication, and Web pages that are used to communicate information to a larger audience. Traditional methods for the exchange of information in the scientific community have been through conferences, seminars, journals and personal communication. While these methods of communication are useful, they are not very efficient when dealing with large data files that numerical solutions produce. These large data files contain a variety of information, which can be displayed a variety of ways. The use of small figures and graphs in journal papers and during conference presentations is not a very effective way to convey the entire solution. Because of size and time limitations, large numerical results are drastically compressed into a few simple figures. The results that are presented to other interested parties, therefore, are usually incomplete.

With ARCOS, complete computed solution data files can be made available for interested parties to download. By providing entire numerical data files instead of a select number of graphs and tables, researchers will be able to analyze the data more closely and obtain a clearer picture of the entire numerical solution. A clear example of the scientific community not using the Internet effectively is seen in the absence of numerical data files available to be downloaded. A screen shot of the available downloads for one numerical code is shown in Figure 8. This Web page containing the results of a series of runs was automatically generated by ARCOS. It contains not only the output data files, but also the input data files and the time that this computation was performed. ARCOS allows interested users access to numerical codes, and more importantly, allows them to execute these codes using their own initial parameters, and then the ability to download the output data files to examine for themselves. This open approach to numerical code development is beneficial to both the developer and the user.

## Accurate Evaluation of Numerical Methods

ARCOS is the most advanced method to effectively evaluate numerical codes. In many cases, different researchers will have their own numerical code which solves a similar problem. Researchers may have also developed a new solution method with which to solve a problem. They will need to compare their own computational results of some certain known condition with solutions that have been produced by other numerical codes. This validation procedure is standard practice in the development of numerical computer codes, as it is used as a check to ensure that the numerical code is computing the correct solution. Additionally, it is regular practice now for journals to ask that numerical results that are presented must include an analysis of the numerical accuracy [15]. Currently a clear validation is very difficult to achieve. Validation test cases are obtained from journals and conference papers. Results presented in these papers, however, may be incomplete, may be displayed using a small figure, or may be the result of unknown initial conditions and unknown methods. This makes it very difficult for an accurate comparison to be made to determine if the numerical code is computing the solution accurately and correctly. The only way in which a clear comparison could be made is if both numerical codes were executed with the same flow conditions, and the output data compared.

ARCOS improves the current validation procedure in two significant ways.

- The system can provide existing solution data files for interested parties to download and examine for themselves. These data files can be used to compare to data that is produced by the researchers themselves, and a conclusion drawn about the validity of their current methods. By providing entire data files, researchers will be provided more accurate information than what is presented in papers and journals. Additionally, not only can computationally computed data files be placed on the Internet for validation purposes, but experimental data files as well.

- The other way ARCOS will improve the validation procedure is for interested parties to remotely execute numerical codes using initial conditions that they desire. By enabling outside parties access to the numerical codes, they can now choose what initial conditions they wish to study. This method ensures that the data that is generated has been produced using well defined initial conditions.

Presently, it has not been common practice for numerical code developers to allow public access to their programs and their solution data files. ARCOS, however, does not make the source code visible, only the input and output data files. The benefits of open communication are much greater than any risk. With no access to source codes, or even the executable, there is virtually no risk to the developers and their codes. Scientific research has always proceeded in a manner where independent analysis was not only required, but desired. Computational research, however, has not involved independent analysis for many significant reasons. This has resulted in the slow development of new methods and codes, which ARCOS can alleviate.

## Graphical Capabilities

ARCOS not only allows users to remotely execute numerical codes and download data files, but it also allows remote users to immediately view these data files in a graphical manner. The output of each separate numerical code is in a different format, and for a remote user that has downloaded a data file, it may take time to configure the data file into a format where it can be plotted. ARCOS has a built in graphical capability comparable to post-processors that are available. The only restriction on using the graphical window is that the code developers must configure the output of their programs to follow the preset format. Once this has been done, a remote user does not need to download a data file, they can immediately view the data in a variety of ways using a Java applet embedded in a Web browser.

There are three different plots that ARCOS is able to generate. The first plot is of the numerical grid. Every physical process that is computationally simulated requires some sort of grid. ARCOS is capable of displaying results for grids ranging from one-dimensional to three-dimensional geometries. Once the grid is displayed within the graphical window, the user has many options with which they can manipulate the view. The user may zoom in or out, move the figure around the window, or rotate the figure along all three axes. The controls for the manipulation of the figure are located beside the graphical window. Additionally, the user will have mouse capabilities, where the mouse itself can be used to move and rotate the figure. A figure of a three-dimensional grid is shown in Figure 9.

The second type of plot that is available is a simple line plot. The user simply has to choose the values that are to be represented along the x and y axes, and ARCOS will automatically render it in the graphical window, along with numerical text indicating the limits of both x and y axes. Additionally, the user can find specific values within the window by pressing the mouse button after pointing the mouse pointer at the desired location. The x and y values that correspond to this location will be displayed in a message area directly above the graphical window. A figure of a line plot is shown in Figure 10.

The third type of plot is a color contour. The user must choose which property they wish to display, and along what parts of the grid, they want to view. Once this is done, ARCOS will display the outline of the grid, and a color contour within the borders of the grid. A contour bar is shown on the left hand side of the graphical window signifying the range that the colors represent. A figure of a color contour plot is shown in Figure 11.

## Availability to Novice Users

ARCOS is designed with a graphical user interface (GUI) that is clear and easy to understand. This feature makes any numerical code easier for anybody to use. With numerical codes, the user usually needs to have a degree of understanding of both the code itself and the methods that are being used. The input of initial conditions and a choice of methods may be unclear for people who are not fully familiar with the code's abilities. By using the GUI, any casual user is able to successfully execute a numerical code without extensive prior knowledge of the code's methods or requirements. The GUI contains information areas, where the developer will detail information that will assist the remote users in properly executing the code. By implementing a code over the Internet, and incorporating a user friendly GUI, the users are asked to input specific parameters concerning the case which they want to study. What results is that this makes the developer's code more useful, since the number of researchers that can successfully utilize it's methods is much larger than if the code were not on the Internet, and not made to be user friendly. People who are interested in solving a particular scientific problem may find that a numerical code that exists on the Internet will be able to compute an accurate solution.

## Conclusions

Work that primarily uses computational simulations is now a large part of scientific research. Computational simulations have allowed scientists to learn about situations that cannot be easily reproduced in an experimental manner. The development of numerical codes and numerical methods, however, has evolved into a largely solitary effort. There is a large communication gap between computational code developers, both in terms of methods and results. Unfortunately, traditional methods of communication are ineffective means with which to present computational data. While most current computational results are still being presented through journal papers and conference presentations, the numerical data is difficult to represent using small graphs and figures. Size and time restrictions also limit the amount of information that can be shared between scientists. To eliminate this communication gap, the Internet must be used as a means of efficient communication.

The objective of this work was to develop a new Internet system that would dramatically improve the way that numerical codes are currently developed, tested, used and presented. Currently, there does not exist an easy method for scientists to place their codes and results on the Internet. The Advanced Remote Code Operating System (ARCOS) has been developed and it is a new method of communication that offers significant advantages to traditional mediums. By creating an easy-to-use system which developers only need to embed within a Web page, researchers from around the world will have instant access to numerical codes and solution data files. With this system, more and more researchers will benefit from the instant communication provided by the Internet, of computational results and methods.

## Appendix

More information about ARCOS can be found at our Web page located at :
**http://cfdlab9.seas.ucla.edu/~theodore/version1.1/**

## Acknowledgements

This research that was applied to developing hypersonic heating codes was partially supported by NASA Dryden Flight Research Center under Grant NCC 2-374.

## References

[1] Cable News Network. *http://www.cnn.com.*

[2] Amazon.com - Earth's biggest selection. *http://www.amazon.com.*

[3] CFD Online. *http://www.cfd-online.com.*

[4] T. K. Lee and X. Zhong. Internet based computational fluid dynamics calculation of hypersonic heating. *AIAA 2000-0678*, 2000.

[5] The source for Java technology. *http://www.java.sun.com.*

[6] The World Wide Web Consortium. *http://www.w3.org.*

[7] The Java Boutique: Free Java Applets, Games, Programming Tutorials and Downloads. *http://javaboutique.internet.com.*

[8] Client/Server Frequently Asked Questions. *http://www.faqs.org/faqs/client-server-faq/.*

[9] The Java 3D Community. *http://www.j3d.org.*

[10] R. Quinn and A. E. Kuhl. Comparison of flight-measured and calculated turbulent heat transfer on the X-15 airplane at Mach numbers from 2.5 to 6.0 at low angles of attack. *NASA TM X-939*, 1964.

[11] R. Quinn and M. Palitz. Comparison of measured and calculated turbulent heat transfer on the X-15 airplane at angles of attack up to 19.0. *NASA TM X-1291*, 1966.

[12] L. Gong, W. L. Richards, R. C. Monaghan, and R. D. Quinn. Preliminary analysis for a Mach 8 crossflow transition experiment on the Pegasus space booster. *NASA TM-104272*, 1993.

[13] C. D. Engel and S. C. Praharaj. MINIVER upgrade for the AVID system, Vol I: LANMIN user's manual. *NASA CR-172212*, 1983.

[14] Java applets for engineering education. *http://www.engapplets.vt.edu.*

[15] Inside the back cover. *AIAA Journal*, 38(12), 2000.

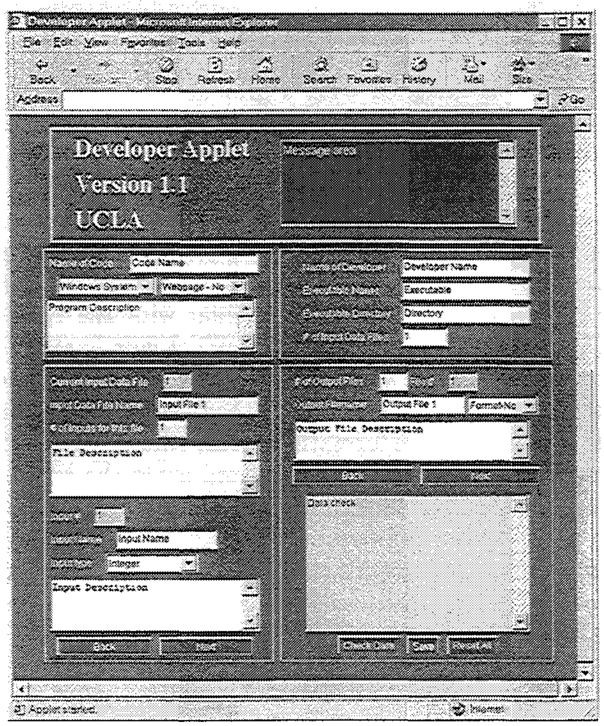


Figure 1: Screen shot of the Java applet used by the code developer.

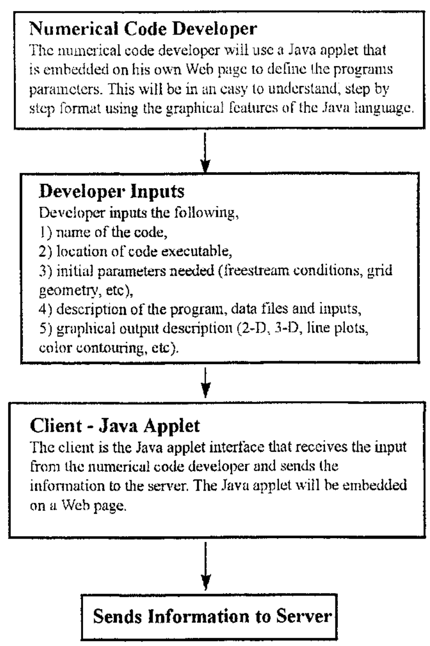


Figure 2: Flow chart of the developer applet.

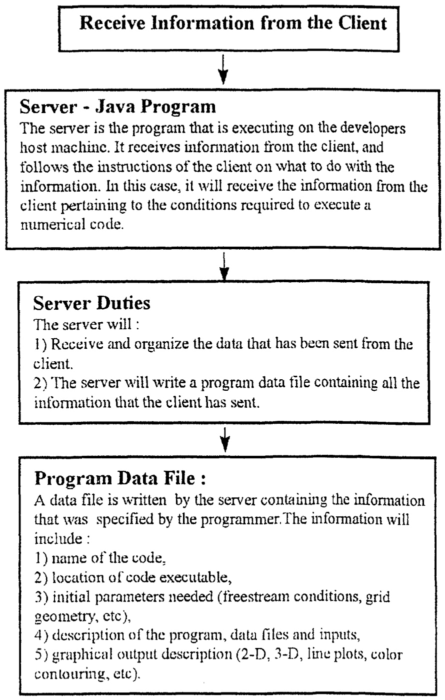


Figure 3: Flow chart of the developer server.

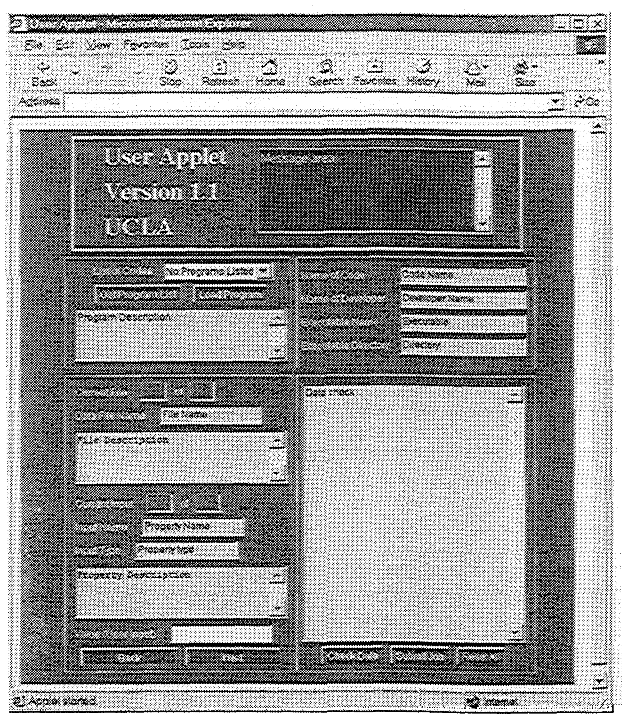


Figure 4: Screen shot of the Java applet used to execute the code.

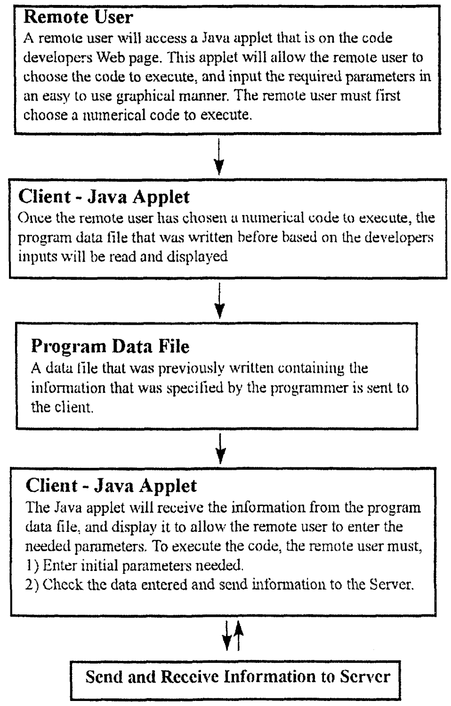


Figure 5: Flow chart of the remote user applet.

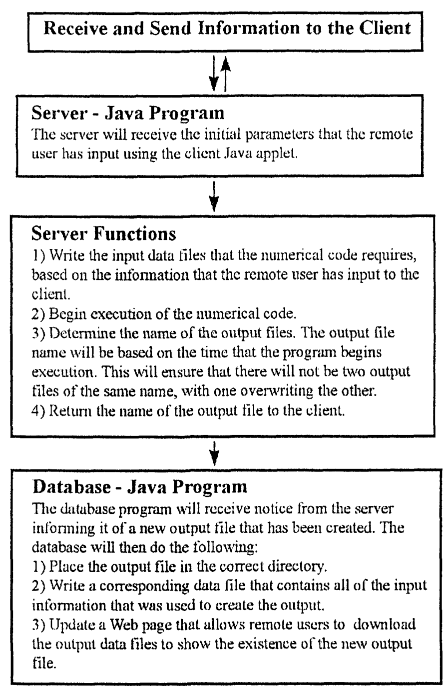


Figure 6: Flow chart of the remote user server.

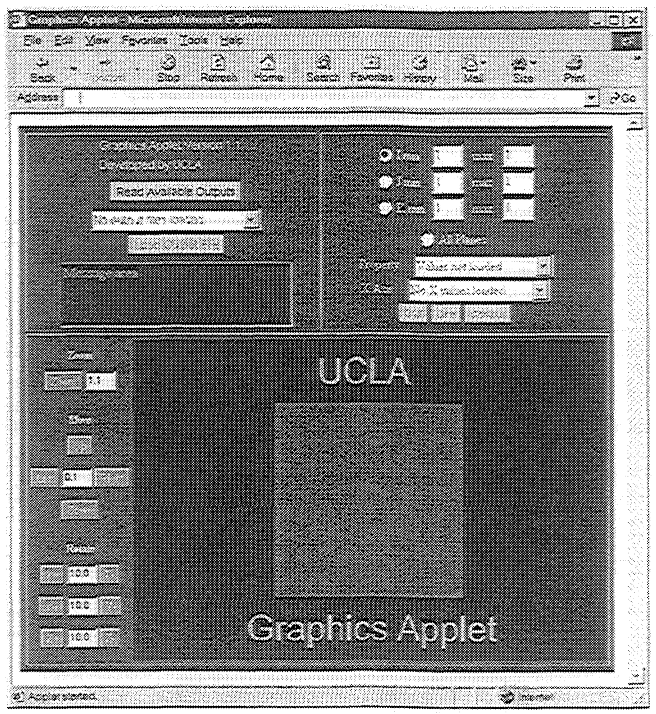


Figure 7: Screen shot of the Java applet used to show graphical plots.

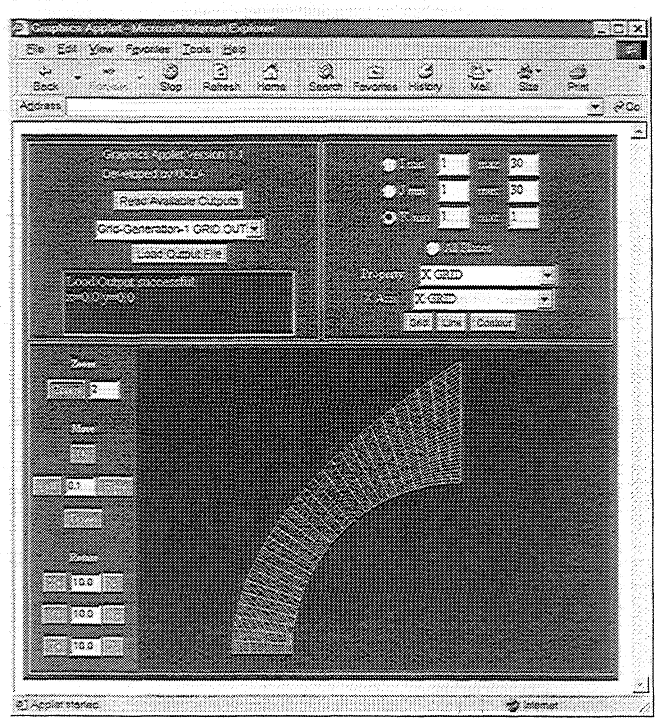

Figure 9: Screen shot of a sphere numerical grid.

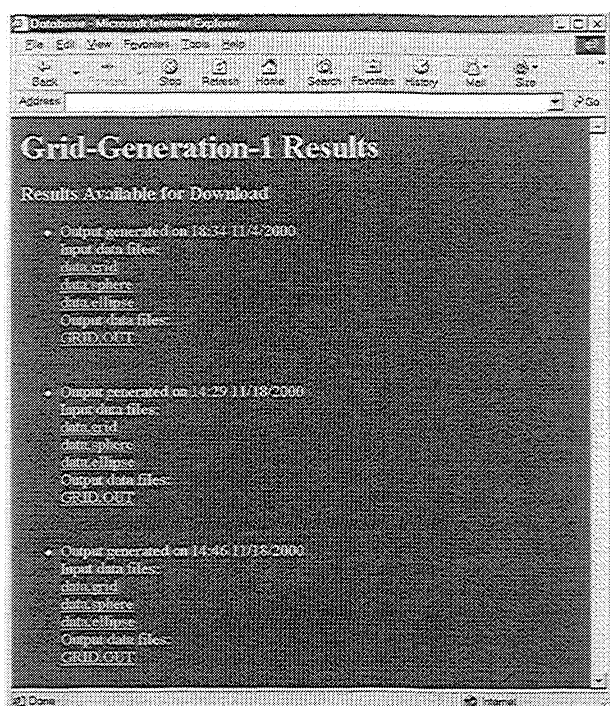


Figure 8: Screen shot of the database of downloadable data files.

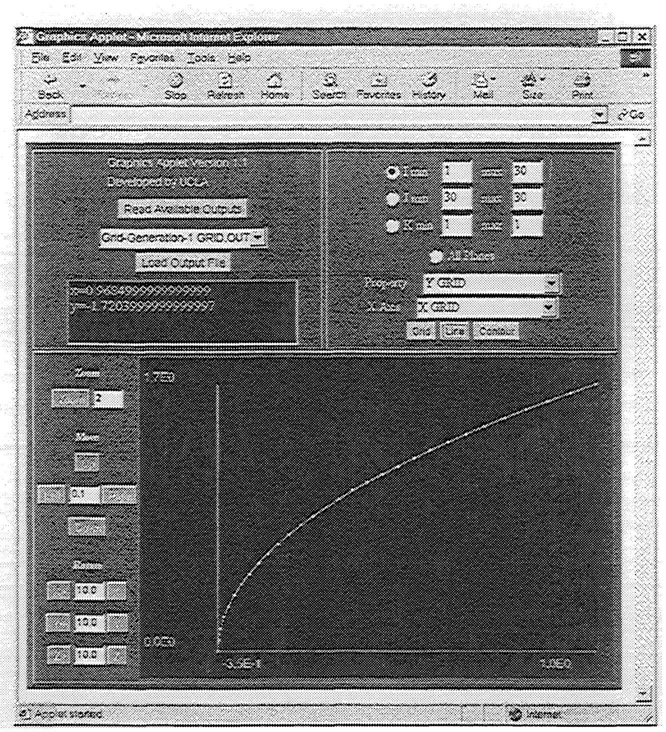

Figure 10: Screen shot of a X-Y line plot.

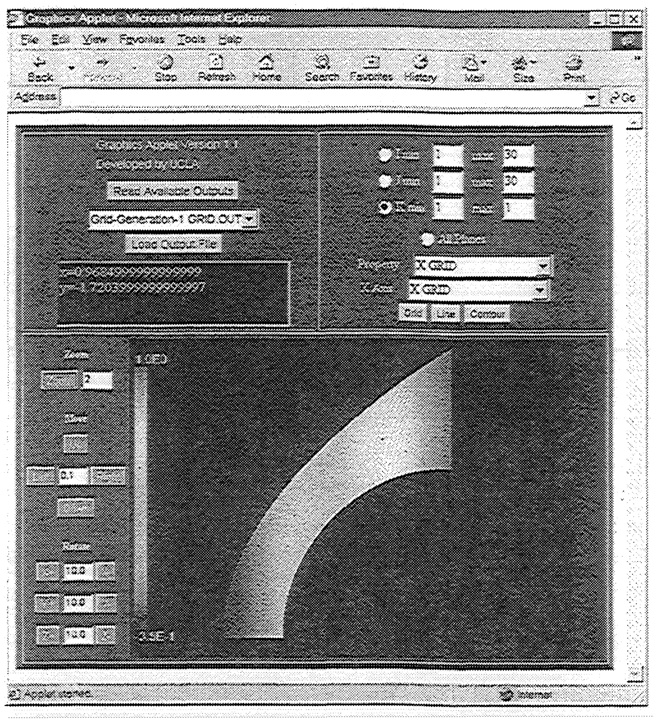

Figure 11: Screen shot of a color contour over a sphere.