Fermi National Accelerator Laboratory

FERMILAB-TM-1896

Xwake 1.0 User's Manual

Greg Saewert, Thomas Jurgens, and Fady Harfoush

Fermi National Accelerator Laboratory
P.O. Box 500, Batavia, Illinois 60510

July 1994

Operated by Universities Research Association Inc. under Contract No. DE-AC02-76CHO3000 with the United States Department of Energy

Disclaimer

Xwake 1.0
User's Manual

June 25, 1994

Fermi National Accelerator Laboratory
P.O. Box 500
Batavia, IL 60510 USA

1.0 INTRODUCTION

Xwake is a user friendly Finite Difference Time Domain (FDTD) code for wake potential and impedance calculations of rotationally symmetric structures. Geometry boundaries are automatically meshed, or modeled, by a choice of either of two approximations, the typical "stepped edge" modeling, or the new "contour" modeling. Contouring means geometry boundaries described with curves and angles are not distorted to follow the rectangular grid; rather, the grid is adjusted to follow the contour of the geometry boundaries. It offers computer run time savings for large problems or intricate geometries for a given mesh compared with stepped edge modeling. Using stepped approximations with Xwake, on the other hand, one is able to model materials of any linear media type. Ultimately, three solutions are computed, wake potential versus wake length, wake spectrum versus frequency—a Fourier transform (FFT) of the wake potential, and wake impedance versus frequency.

Xwake is an OSF/Motif application providing a graphical user interface (GUI) for ease of use. It is written in ANSI C for portability and currently runs on UNIX platforms.

1.1 Using Xwake

There are two stages to using Xwake. Initially the user describes the geometry and boundary conditions in a "problem description" file. This is done outside of running the program itself. The user is free to use any text editor to produce this ASCII, text file. Subsequently, Xwake is executed to make the wakefield calculations and display the results. This involves opening the problem description file, then sequentially running two computing routines, the Mesh Generator and the Solver. Wakefield solutions are computed by the Solver. The user then displays the results in the form of plots on the computer screen.

Xwake includes some features for convenience. Data computed by the Mesh Generator and Solver can be saved in a file. As a debugging aid for the contents of the problem description file, the described geometry can be plotted to the screen. For documentation purposes any plot can be written to a file or routed directly to a PostScript printer.

Effort was made to make Xwake intuitively easy to use and includes error checking to make it robust. For example, only those menu options which allow valid actions at any point in time are enabled. It is not possible to do anything "wrong" or out of sequence. When necessary, informative messages are displayed. You should find this program very easy to use.

1.2 Included In This Guide

Section 2, CAPABILITIES, elaborates a bit on Xwake version 1.0's capabilities .

Section 3, RUNNING XWAKE, explains how to use the program. Section 3.1 provides of an overview, while Section 3.2 lists the instructions to walk through the program from start to finish. Sections 3.3 and 3.4 serve as a reference; they explain every option available on the menu in detail.

Section 4, THE PROBLEM DESCRIPTION FILE, explains how to write a problem description file from scratch. Explanation is given by way of an example including a drawing, file listing and line-by-line explanation of the file. Useful reference information is provided in Sections 4.4 and 4.5 which include tips and a glossary.

Section 5, INSTALLATION, explains how to obtain a copy of Xwake and install it on a UNIX workstation.

2.0 CAPABILITIES

Xwake is an 2.5 dimensional electromagnetic code written for wake field and impedance calculations of rotationally symmetric structures. This dictates using the cylindrical coordinate form of the relevant electromagnetic field equations. The azimuthal dependence of these equations are expanded in a sum of sine and cosine modes. The radial and axial (r and z) dependence of the fields are solved for by the code for a given azimuthal mode number. Xwake version 1.0 permits modeling with azimuthal mode number, m = 0, only. For m = 0, computed solutions have no variation in the ϕ direction.

With Xwake, a user may describe an complicated region that, if needed, may be multiply connected. The user is given the ability to visually display any described problem and obtain a hardcopy for documentation.

The code also permits modeling region boundaries with either a conformal FDTD approximation or an older stair-stepped FDTD approximation. The stair-stepped FDTD approximation results from the differencing of Maxwell's curl equations. The conformal approximation employs an FDTD discretization of Ampere's and Faraday's laws[1,2] which deforms grid cells that are only near a region boundary. Cells that are not near a region boundary possess a rectangular shape whose difference equation is identical to that of a stair-stepped approximated cell. Therefore, the cost of computing the irregularly shaped region boundary cells has negligible impact on Xwake's computer resource needs. In addition, Xwake has automatic dynamic memory management which enables the user to specify problems that make full use of available computer resources.

When run with the conformal approximation option, Xwake 1.0 permits the accurate modeling of curved and sloped material boundaries. Fine geometries can be modeled without resorting to the computationally costly practice of decreasing the discretization mesh size. This means that structures that are less than the grid discretization such as thin plates and small bumps can be incorporated into the model at negligible computing cost. With the conformal option, the current version of the code only allows the use of vacuum and metallic materials.

With the stepped approximation chosen, Xwake 1.0 can model structures of arbitrary composition. The conductivity, permeability and permittivity of a material may be any physically realistic, scalar, constant value. The material complexity of the structure has no effect on Xwake's use of computer resources.

3.0 RUNNING XWAKE

After explaining some basics in Section 3.1, Section 3.2 will take you through the program from start to finish, step-by-step. Features not covered in these two sections are thoroughly explained in Sections 3.3. and 3.4 where every menu option is explained.

3.1 Overview

The flow of the program control is illustrated in Figure 1. Producing wakefield data once a problem description file has been created is a three step process. Each of these steps produce a set of data. Opening a problem description file produces *Prob* data. Running the Mesh

[1] Conformal Modeling of 3D Wake Fields, Proc. 14th Particle Accelerator Conf., pp 321-323, 1991.

[2] Finite Difference Time Domain Modeling of Curved Surfaces, IEEE Trans. Antennas Propagation, vol. 40, 1992.

Generator (MG) on *Prob* data produces *Mesh* data. Running the Solver on *Mesh* data produces *Soln* data. When data has been produced it is said to be "loaded". The three shaded boxes in Figure 1 represent the three loaded data sets, *Prob*, *Mesh* and *Soln*. Notice that *Prob* data can only be loaded by opening a *Prob* data file. *Mesh* and *Soln* data, on the other hand, can be loaded by running the MG or Solver, respectively, or by opening a previously saved data file. Rectangles represent control options found on menu bars. Ellipses represent files on disk.

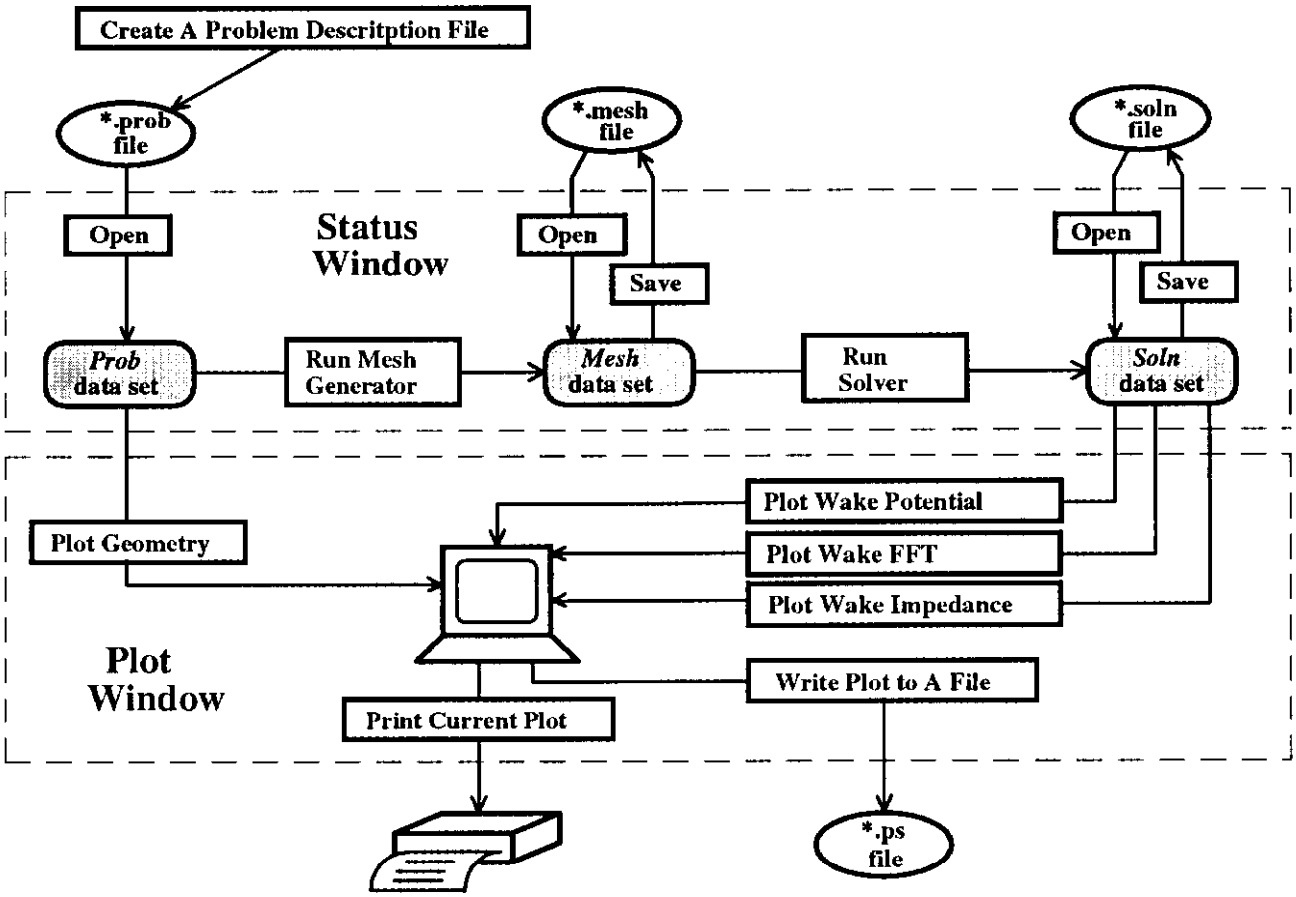


Figure 1. Program Control Flow Diagram

Figure 1 shows that Xwake's controls are divided between two main windows, the Status Window and the Plot Window. Each has its own display area and controls menu bar. With the Status Window you control and monitor the loading and saving of data sets in the process of arriving at wakefield solutions. With the Plot Window you view the geometry and wakefield results, having the ability to obtain hardcopies of the plots.

3.1.1 Computing Wakefields

In the process of opening a *Prob* file, line segments are calculated from the intersections of the geometry boundary lines and the grid. This information is stored as *Prob* data. Meshing determines which region each cell in the grid belongs to and assigns the appropriate media characteristic to it. At the time the user meshes the problem is when they choose the boundary

approximation type, stepped edge or contour. Meshing can require reasonable run time, depending on the size of the problem and the granularity of the grid. It is advantageous, therefore, to be able to save *Mesh* data. It will save time meshing again if you run the Solver on the problem in the future. All field calculations are done in the Solver and require *Mesh* data to be loaded. The *Soln* data file contains only the results of a run—wake potential, FFT of the potential, and impedance of the wake. Explanations on running the MG and the Solver are found in Section 3.3.2.

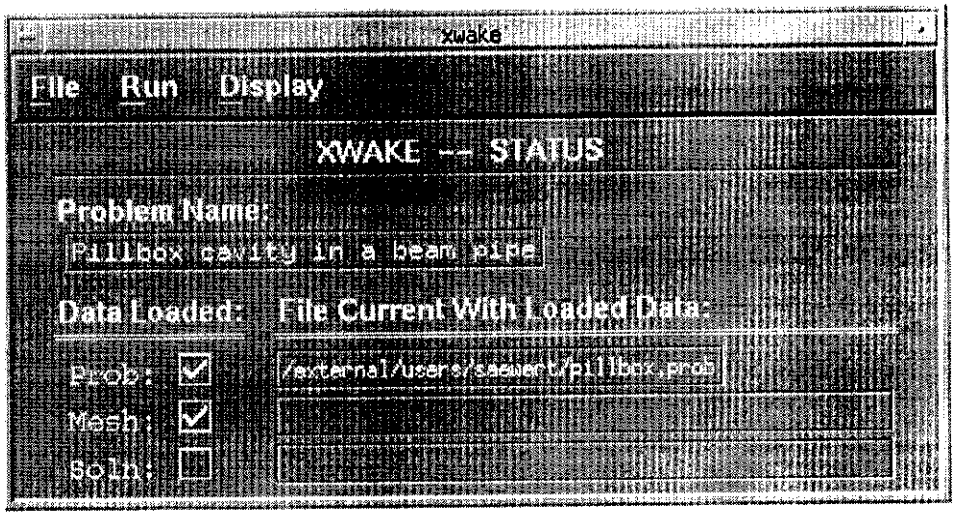


Figure 2. Status Window

A picture of the Status Window is shown in Figure 2. As explained in Section 4, when creating a problem description file a problem name must be chosen and placed on line one of the file. This name is treated as part of the data, so when a data file is opened, regardless of the type, the associated problem name is read in and displayed under the title "Problem Name:". Figure 2 illustrates that a file for a problem named "Pillbox cavity in a beam pipe" has been opened.

Also revealed in Figure 2 is that file names appear next to each data set type. The name displayed will be either the name of a file which has been opened, or the name of a file to which a data set has been saved. Furthermore, it will only be displayed if it can be assured that the contents of the file corresponds exactly to what is loaded. For example, suppose you run the Solver, save the *Soln* data, then run the Solver again. The name of the *Soln* file will be cleared away because the Solver may have been run with different parameter values the second time, and Xwake cannot be certain that the file whose name was displayed is still current with the *Soln* data now loaded. Figure 2 shows that it was a *Prob* file named "pillbox.prob" that was opened.

Figure 2 also shows that the MG has been run but the Solver has not. Also evident is that the *Mesh* data has not been saved to a file. Section 3.3 describes each control option on the Status Window's menus.

3.1.2 Viewing Results

Figure 4 is a picture of the Plot Window. Again, the plotting options include the ability to draw the described geometry in loaded *Prob* data, as well as the wakefield solutions. It is

recommended you plot the problem geometry prior to running the MG for the first time in order to check for errors. Any currently viewed plot can be dumped to a PostScript file or sent directly to a printer.

There are two ways to preserve results from Xwake. You can save *Soln* data set files or printouts of the plots. Section 3.4 explains each control option on the Plot Window's menus.

3.2 Walk Through Xwake Step-By-Step

The following will describe a direct route through Xwake assuming a problem description file is available. Since there are several sample description files that accompany Xwake, it is not necessary to write one from scratch in order to run this program for the first time. If you do not have these sample description files see Section 5 on how to obtain them.

I) The first step in computing wakefield data is to create, or at least identify, a file describing the geometry of some problem. We will assume that one of the sample files is available.

II) The next step is to produce wakefield solution data. Do the following:
   1) Start Xwake from the local directory containing the file(s) identified in I).
   2) Open one of the example problem description files.
      A) Click the following three button sequence: **File | Open | Prob**.
      B) You are presented with a file selection dialog. Click once on the name of one of the files.
      C) Click **OK.**
      D) Observe that a check mark appears in the Status Window next to *Prob* indicating that *Prob* data is loaded.
   3) Run the MG to mesh the problem's region boundaries with the described grid.
      A) Click this two button sequence: **Run | Run Mesh Generator**.
      B) A dialog appears with two options, Stepped and Contour. Section 2 explains the significance of these two choices. Section 5 will explain which approximation is preferred for the file you've chosen.
      C) We can accept the default, Stepped Edge, so just click **OK** to initiate the Mesh Generator.
      D) A message dialog appears while the MG is running. When completed a check mark appears in the Status Window indicating the *Mesh* data is loaded.
   4) Run the Solver.
      A) Click this two button sequence: **Run | Run Solver**.
      B) A dialog appears containing 6 Solver parameters. (The explanation of the **Run Solver** button in Section 3.3.2. provides an explanation of these parameters.)
      C) For now, we'll accept the default values, so just click **OK** to run the Solver.
      D) A message dialog appears while the Solver is running. When completed a check mark appears on the Status Window indicating *Soln* data is loaded.

III) The third step is to plot wakefield data on the screen.
   1) Open the Plot Window by clicking the two button sequence: **Display | Open Plot Window**.
   2) There are four options on the **Plot** menu of the Plot Window menu bar available, Problem Geometry, Wake Potential, Wake Frequency Spectrum, and Wake Impedance. Select each one of these options.
   3) It is at this point after viewing plots that you would decide to save results of a Solver run. You have several options; you could save the *Soln* data file, you can send a plot directly to a PostScript printer, or you can save the plot in a PostScript file. The

option to save *Soln* data is available under the Status Window's **File** menu and explained in Section 3.3.1 The other two options are available under the Plot Window's **Hardcopy** menu and explained in Section 3.4.2.

IV) Exit the program.

1) From the Status Window choose: **File | Exit**.

A) You are presented with an option because there is *Mesh* and *Soln* data loaded that have not been saved to a file. Xwake prevents you from inadvertently losing costly calculations.

B) Choose **YES** to exit.

You may have noticed a couple of option buttons on menus that were not described, above. You will find their explanations in the following sections along with explanations for those you did encounter.

3.3 The Status Window's Menu Bar Controls

The following is a explanation of the individual menu options on the Status Window's menu bar.

3.3.1 The "File" menu:

Open This button has three options, **Prob**, **Mesh**, and **Soln**, corresponding to the data set types. Selecting one brings up Motif's standard File Selection Dialog with a list of files in the current directory filtered to the particular type of data chosen. Having selected the **Prob** button on the menu bar will filter out those files of the type, "*.prob", for example.

Tips on opening files:

- Choose a file in the list under, "Files", by double clicking on a file in the list, or, by clicking on one of them once and then clicking on **OK**.
- Choose a file not in the list by entering the known file name under the heading, "Selection", then click **OK**.
- A file can be opened at anytime (except while the MG or Solver are running).
- Opening a file will unload all currently loaded data sets, but not without first giving you the option of saving any existing *Mesh* or *Soln* data if they have not been saved.
- With the keyboard focus on the Status Window, you can bring up the file selection dialog to open a *Prob* data file directly by hitting <Shift>P; a *Mesh* file by <Shift>M; or, a *Soln* file by <Shift>S.

Save This button has two options, **Mesh** and **Soln**. (*Prob* data is so readily produced from the description file that it is not offered as one of the options to save). Choosing to save a data set brings up the File Selection Dialog. To save a file for the first time enter the full name of the file in the box under the heading "Selection", then hit **OK**.

Tips on saving files:

- To facilitate finding data files in the future, it is strongly suggested to follow the convention of choosing extensions for file names matching the data type being saved, i.e., .prob, .mesh or .soln.
- Saving *Mesh* data enables later wake analysis by the Solver, since the Solver needs only *Mesh* data to be loaded in order to run.
- Saving *Soln* data only saves the results of a single Solver run.

- With the focus on the Status Window the file selection dialog for saving *Mesh* data can be popped up directly by hitting <Ctrl>M, to save *Soln* data by <Ctrl>S.

About Xwake Along with displaying the version of Xwake this shows where to obtain Xwake on Internet and who can be contacted for questions or comments.

Exit This allows you to exit the program. Should there be any *Mesh* or *Soln* data sets that have not been saved to a file, you are presented with an option to either go back or continue to exit.

3.3.2 The "Run" menu:

Mesh Generator This button is enabled when *Prob* data is loaded. Actuating this button brings up a dialog for selecting the meshing boundary approximation type, stepped edge or contour. Explanation of these two types is provided in Section 2, Xwake Capabilities. **OK** initiates the Mesh Generator. The MG requires time to compute so a progress message is presented which contains an **ABORT** button.

Tips on running the Mesh Generator:

- The size of the meshing task also effects the update rate of the progress message; and, acknowledgment of an **ABORT** button press occurs only at the time the message is updated. So, for a large problem you will notice a delay between pressing this button and the time the abort actually occurs.
- With the keyboard focus on the Status Window, the Run Mesh Generator dialog can be popped up directly by hitting F1.
- Occasionally the contour mesh generator will not be able to grid a problem. This is usually caused by a problem description file that has an intricate geometry that is much smaller than a grid dimension, making the grid finer will alleviate this condition.

Run The Solver This button is enabled when *Mesh* data is loaded. Actuating this button brings up a dialog containing 6 parameters, appearing in editable boxes, required for computing wakefields. Figure 3 shows this dialog. Default values are initially calculated from the information in the problem description file and appear in the boxes. You can accept these values or change any of them before running the Solver. **OK** initiates the Solver and accepts the parameter values currently displayed. **CANCEL** exits without doing or changing anything, and **DEFAULTS** reinstates the initial default values. As with MG, the Solver requires time to compute so a progress message appears containing an **ABORT** button.

Solver Parameters:

Bunch length (rms): The rms length in meters of the Gaussian shaped beam. The default value is equal to 10 times dz.

Azimuthal mode number: Rotational symmetry. Currently, Xwake can be run for mode = 0 only.

Radius of hollow beam: Radius of ring-like beam with assumed azimuthal symmetry. The only allowed value, the default, is rpipe—described in the problem description file.

Gaussian cut-off (rms): Number of rms lengths from the center of the Gaussian beyond which the beam is set to zero. The default is 5. The leading edge of the beam is located one Gaussian cut-off in front of the centroid of the Gaussian beam.

Wake length: Length behind the beam's leading edge where the wake field is integrated over the total length of the structure. The default is equal to 10 times the bunch length.

Radial wake offset: Radial position where wake fields are integrated. The only allowed value, the default, is rpipe.

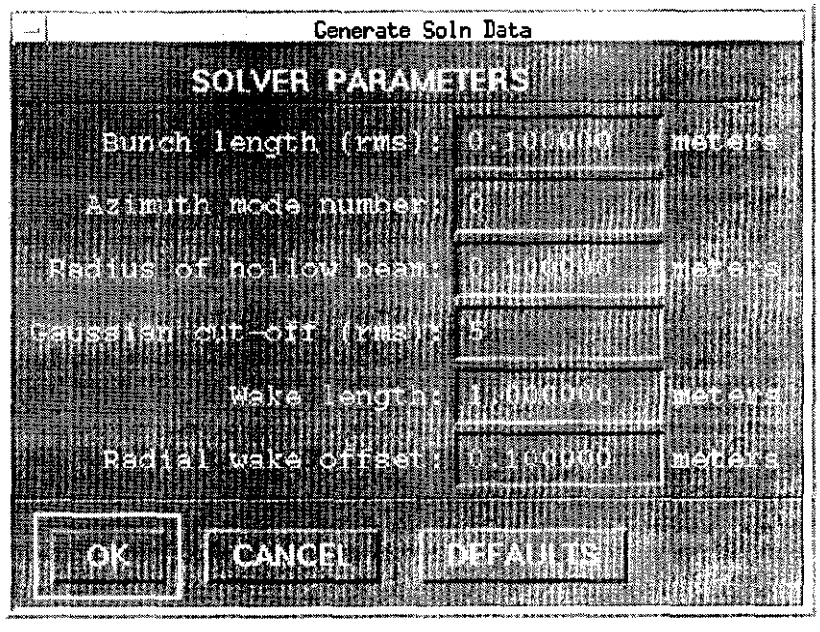


Figure 3. The Solver Parameters Dialog

Tips on running the Solver:

- Running the Solver requires only that *Mesh* data be loaded—*Prob* data is not required. Therefore, you can run the Solver after only opening a previously saved *Mesh* data file.
- The Solver completion date and time are saved with the data. This date appears on the wakefield data plots.
- The factors affecting the Solver's calculation time and the **ABORT** button response time are much the same as for the MG.
- With the keyboard focus on the Status Window, the Run Solver dialog can be popped up directly by hitting F2.

3.3.3 The "Display" menu:

Open Plot Window *This button allows opening the Plot Window at any time whether* there is currently any data available for plotting or not. With the keyboard focus on the Status Window, the Plot Window can be popped up directly by hitting F3.

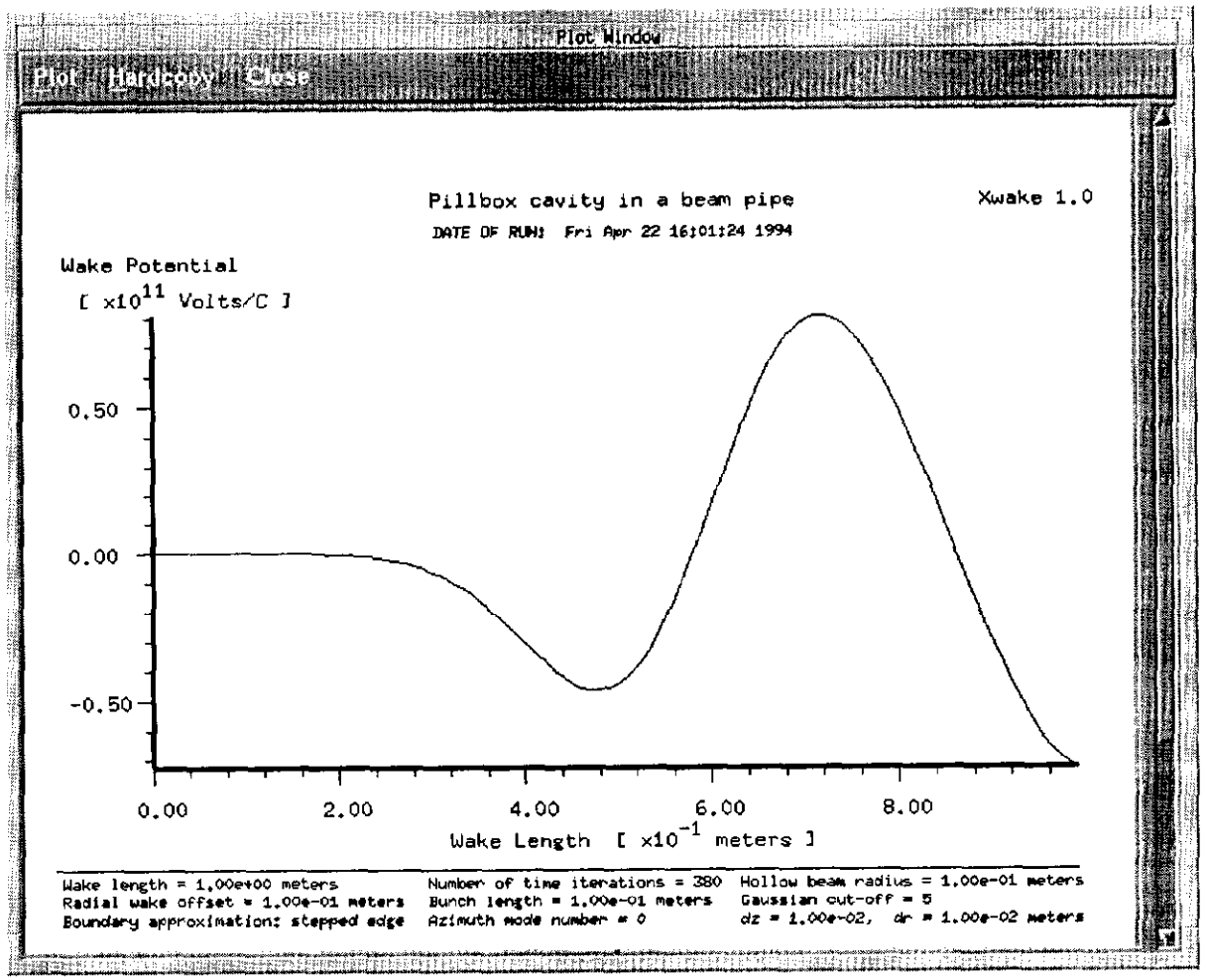


Figure 4. Plot Window

3.4 The Plot Window's Menu Bar Controls

Figure 4 shows the Plot Window. This window is not permanent. It can be opened and closed at any time. The following explains the menu bar controls:

3.4.1 The "Plot" menu:

Problem Geometry *Prob* data must be loaded to view the described geometry. Three options are available to assist in resolving the various aspects of the described geometry. Aspects of the geometry are the region outer boundary lines, problem boundary lines or not, and the meshing grid. Choosing **All Regions & Grid** displays all the above. Problem boundary lines are plotted with thicker lines than the region outer boundaries. If a region outer boundary lies on a grid it will be hidden by the lines of the grid when viewed on the screen. (PostScript output plots, however, use gray scale to distinguish grid lines from region outer boundaries.) Also, if the grid is defined to be very dense this plot may appear almost black. To avoid these difficulties choose the **All Regions, No Grid** option which will not show the grid. If there are

errors in the description of the geometry the pattern of lines might be confusing. Therefore, **Boundary Only** is an option which will plot only those lines defined as problem boundary line segments. If described correctly, this plot should appear as a closed rectangle drawn with the thick line style. The date and time appearing on the geometry plot corresponds to the time the plot was produced.

Wake Potential This displays the wake potential plotted in units of volts per coulomb versus wake length.

Wake Frequency Spectrum (FFT) This is a Fourier Transform of the wake potential. Units are in volts per coulomb versus frequency. The scaling of this plot depends on settings of the **Scaling Options**.

Wake Impedance This displays the impedance of the wake potential computed by convoluting the wake FFT with an input impulse function. It is plotted in units of ohms versus frequency. The scaling of this plot depends on settings of the **Scaling Options**.

Tips on plotting the above *Soln* data:

- To plot any of the wakefield data, only *Soln* data has to be loaded. Therefore, there are several scenarios for making wakefield data available to plot:
  1) Start with opening a problem description file (which loads *Prob* data), run the MG, then run the Solver. *Soln* data is available to plot.
  2) Open a previously saved Mesh data file. Then run the Solver.
  3) Open a previously saved Soln data file.

Scaling Options This presents a dialog with options effecting the scales of the FFT plot and impedance plot. The options include magnitude, real, magnitude and phase or real and imaginary components for the data. Also, the frequency scale can be set to either linear or logarithmic. The increments on the log scale are multiples of 1, 2, 4, 6 and 8. This dialog can remain up as long as desired.

3.4.2 The "Hardcopy" menu:

As illustrated in Figure 1, this menu contains two options for producing hardcopies, to print the currently view plot directly to a PostScript printer or to write the plot to file (PostScript format). Before a plot can be sent to the printer, the user must configure the printer. These three options are further explained:

Write Plot To A File This button pops up a dialog to enter the name of a file in which to save the currently viewed plot. The dialog contains an **OK** and **CANCEL** button. **OK** writes the plot to the chosen file.

Tips on producing hard copies of plots:

- The file name selection dialog remembers the last name you entered facilitating successive plot dumps. You only need to make small changes to the name on each successive dump.
- You will overwrite the previous file if you do not change the name on the next dump.

Print Current Plot This option is only available after a plot is on the screen and the printer has been configured. The plot is temporarily placed in a file before a command is issued to UNIX to print. This file is automatically deleted at program termination.

Configure Printer This presents a dialog in which you enter the appropriate UNIX command for routing a file to the desired PostScript printer. Entered the UNIX command, only, without the name of a file. The printer must be configured before **Print Current Plot** is enabled. This information is saved in a file in the user's root directory.

3.4.3 The "Close" menu:

Close Plot Window This will close the Plot Window as well as the **Scaling Options** dialog, should it still be popped up. With the keyboard focus on the Plot Window, the Plot Window will pop down directly by hitting <Alt>F10.

3.5 Keyboard Traversal

Being an XWindow/Motif application, the GUI of this program responds to control selections on dialogs with either the mouse or by means of keystrokes. Use of the mouse requires no explanation. However, you can navigate the menus and dialogs without the mouse.

To select a desired menu on either the Status or Plot Windows, first make sure that window has the keyboard focus. Then hold down <Alt> while you hit the key corresponding to the underlined letter in the name of the desired menu. That brings up the menu. To select an option from the menu, simply hit the key corresponding to the underlined letter on the desired menu option.

Dialogs can be traversed to advance keyboard focus by use of the Tab key. The highlighted entity on a dialog has the keyboard focus. When a dialog first appears, the **OK** usually has a box around it indicating it has the focus, for example. When a button has the focus, it can be actuated with the Enter key. You can Tab until the focus is on the desired button and hit Enter to select that button. This can be of benefit on some monitors when the bottom of a dialog is off the screen.

4.0 THE PROBLEM DESCRIPTION FILE

The problem description file, or "problem file", describes the geometry of a rotationally symmetric structure such as a vacuum beam pipe and its surrounding structure. The file format is ASCII and, therefore, can be created with any text editor. For convenience the user should name the file with the extension .prob. Such files are by default filtered out in the File Selection Dialog upon being opened. What follows is a detailed description of the rules and "language" for describing the problem geometry. To help us in this task, an example problem file is presented which demonstrates the use of all the problem file language vocabulary. A diagram of its structure and a file listing are included. A glossary is found at the end of this section.

4.1 File Organization And Concepts

The coordinates used to describe a given geometry are in the usual (r, z) plane and must be dimensioned in MKS units. Any number of materials can be described within the structure, but the minimum a problem file will contain is two—the vacuum space inside the beam pipe and the surrounding conductor. Materials within the structure can have any permeability, permittivity, and conductivity.

The cross sectional surface of each homogeneous material in the structure defines a "region". The outer edge of each region is referred to as the "region outer boundary". Region outer boundaries are described as closed polygons composed of segments: lines, circular arcs and elliptical arcs. A problem file consists of descriptions of regions in such a way that the regions fit together like pieces of a puzzle to form a rectangle. The outer edge of this rectangle is specifically referred to as the "problem boundary" and defines the "problem space". Like a puzzle, within the problem space there must be no areas left undescribed, and regions must not overlap. All polygon segments that lie along the problem boundary must include an extra identifier (explained in Section 4.2.2). That the problem boundary is a closed rectangle can be easily checked when running Xwake by plotting the problem geometry (see Section 3.4). Xwake displays the problem boundary with a thicker line than ordinary region outer boundaries. This is illustrated in Figure 1 of our example problem.

A region can contain "holes". Holes are areas within an, otherwise, homogeneous region that contain material having media characteristics different than the surrounding region. While a region is a description of material, a hole is a description of an area within a region where there is an absence of material. The area described by a hole must, in addition, be completely filled by the descriptions of one or more regions. (Remember our analogy to a puzzle.) The boundary of a hole is referred as a "region inner boundary". A region that has no holes is said to be "simply-connected". Regions containing holes are said to be "multiply connected". Looking at our example, Region 2 is a multiply connected region. There is a single hole in Region 2 which contains Regions 3 and 4. Regions 1, 3 and 4 are simply-connected regions.

Listing 1 was put together to demonstrate all of the features which can be used to describe a problem file. Because it contains materials other than perfect conductors and vacuum it would be inappropriate to mesh the problem with the contour approximation method, as mentioned in Section 2.0.

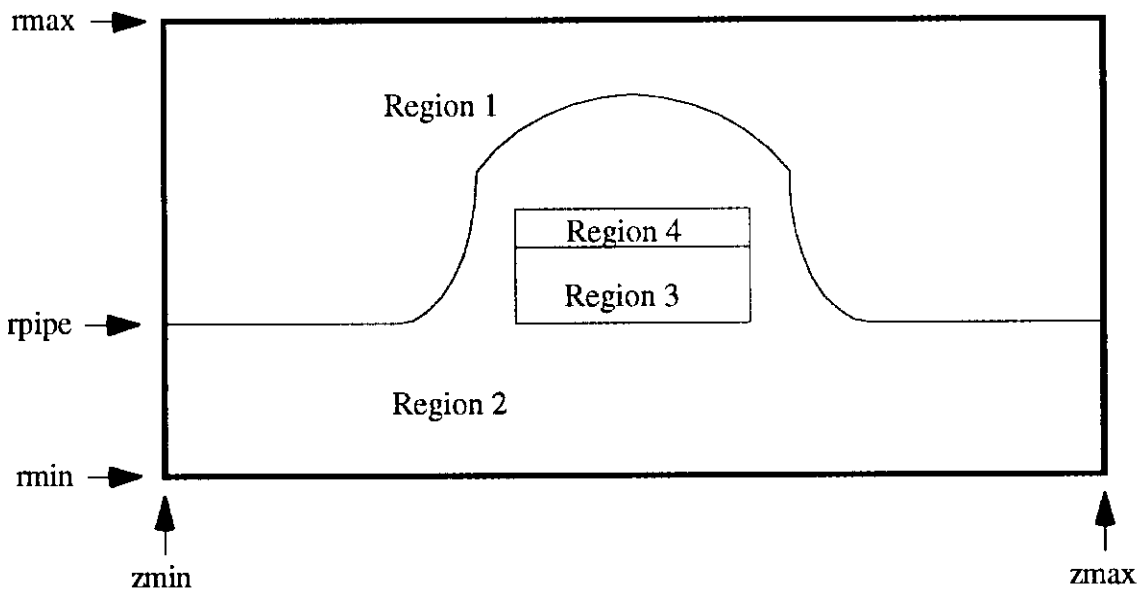


Figure 1. Example Problem Geometry

4.2 Syntax

The first two lines in the problem file are header lines followed by descriptions of regions and holes. The very last line in the file must contain the one word, "end". Comment lines can be included anywhere in the file (see the Glossary for the syntax of the "#" symbol).

Assign a number to every region in the file starting with "1" and incrementing sequentially. If a region is multiply connected, follow the description of the region, immediately, with the description of the holes in that region. Holes are considered part of a region's description. Similar to regions, holes are assigned numbers and must increment sequentially starting with "1". If more than one region has holes, the numbering of the holes in each region must begin with "1".

4.2.1 Header Lines

The first line in the header contains the name you choose for a title. The title including blank spaces should not exceed 50 characters.

The second line defines values of the mesh and pipe radius. The parameters, **dr=** and **dz=**, define the mesh step size in the r and z directions, respectively. The total grid dimensions are specified with the parameters, **rmin=**, **rmax=**, **zmin=** and **zmax=**. Refer to Figure 1 for their meaning. The beam pipe radius is specified with **rpipe=**. Listing 1 illustrates the syntax of these seven parameters.

Syntax Rules:

1) A space or tab must appear between the "=" sign and the value.
2) A space must separate each parameter value and the next parameter name.
3) The total grid size dimensions must coincide with the **line** segments identifying the problem boundary.
4) Every problem file must include all the seven parameters, and they can appear in any order on line 2.

4.2.2 Describing Regions and Holes

Figures 2 and 3 illustrate the general syntax for describing regions and holes, respectively.

Syntax Notation For Figures 2 and 3:

bold parameter names which must be typed explicitly as shown.

italics parameter values you provide:
- those in capital letters, i.e. *N, M*, are integers.
- those in small letters are real numbers (in meters).
- *r* and *z* are the point coordinates in r, z planes respectively.

[ ] optional parameters and values.

(no) the default **bound=** value at the beginning of each region description if not explicitly stated and remains in effect until set to 'yes'.

Line Number Within The Region Description	Region Description Format
line 1	**region** *N*
line 2	**epsilon=** *value* **mu=** *value* **sigma=** *value*
line 3	**start** *r z*
line 4 ⋮	*r z* [**line** [**bound=** [yes / (no)]] / **circ=** *radius* / **ellip=** $radius_r$ $radius_z$]
line i	**fini** [**line** [**bound=** [yes / (no)]] / **circ=** *radius* / **ellip=** $radius_r$ $radius_z$]
line i + 1 ⋮	[**region** *N* **hole** 1]

Figure 2. Region Description Format

Region Description:

line 1 The first line of a region description identifies which region is being defined. The first region described must be "1", and subsequent region numbers must increase sequentially.

line 2 The second line defines the material characteristics for the region. See the glossary for defining the three parameters.

line 3 This line contains the word "**start**" and followed by the coordinates of a point selected arbitrarily on the region's outer boundary.

line 4 This begins with the coordinates of the point which is to be joined with the point on line 3 to form a segment, and includes the parameter indicating the type of connection to be used. There are three types of connections to choose from, lines (**line**), circular arcs (**circ=**) and elliptical arcs (**ellip=**). Refer to the glossary for their syntax. If the connection is a **line** segment and lies on the problem boundary, then "**bound=** yes" must be included.

line 4+ Repeat the syntax of line 4, above, for subsequent segments to complete the description of the polygon.

- You can describe region outer boundary polygons clockwise or counter-clockwise around the region.
- **Line** segments lying on the problem boundary must be identified by including "**bound=** yes" following the coordinates.

- Only **line** segments can be identified to lie on the problem boundary.
- Each line needs to indicate the type of connection to make a segment. However, to help minimize typing, each of the three connection types and the **bound=** parameter stay in effect within one region description until redefined. See Listing 1 which illustrates this optional feature in the Region 2 description.
- "**bound=** no" will turn off any previous "**bound=** yes" declaration.

line i The syntax of this line is used to close the polygon.
- Connection type and **bound=** parameters stay in effect unless changed.

line i+1 This illustrates that for a multiply connected region there will be descriptions of one or more holes following the description of the region's outer boundary.

Line Number Within The Hole Description	Hole Description Format
line 1	**region** *N* **hole** *M*
line 2	**start** *r z*
line 3	*r z* [**line** / **circ=** *radius* / **ellip=** $radius_r$ $radius_z$]
⋮	
line i	**fini** [**line** / **circ=** *radius* / **ellip=** $radius_r$ $radius_z$]

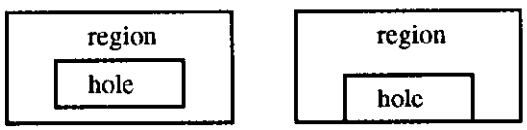
Figure 3. Hole Description Format

Hole Description:

Figure 3 shows the syntax for describing holes. There are three differences in the syntax between hole and region descriptions:

1) Line 1 of the hole description now includes the hole number in addition to the region number.
2) There is no line to describe the material characteristics for a hole.
3) The **bound=** parameter is not valid for **line** segments.

There is one stipulation about holes: no segment of a region's inner boundary (the boundary of a hole) can coincide with that region's outer boundary:



Correct **Incorrect!**

This also means that a **line** segment of a hole will never appear on the problem outer boundary and explains why the **bound=** parameter is not valid. Note that the "incorrect" geometry configuration can be described, however. There is no need for describing a hole. The area occupied by the hole should simply be described as another region.

4.3 Example File

Below is the listing for the example problem illustrated in Figure 1. After the listing is a line-by-line explanation of the listing. (Line numbers have been added for illustration purposes only.)

Listing 1.

```
1   Section 4 Example Problem
2   rmin= 0 rmax= .6 zmin= 0 zmax= 1.2 rpipe= .2 dr= .012 dz= .012
3   #
4   # This listing demonstrates Xwake 1.0's use
5   # of every available parameter.
6   #
7   region 1
8   mu= 1 epsilon= 1 sigma= 1e7
9   start .2 0
10  .6 0 line bound= yes
11  .6 1.2 line bound= yes
12  .2 1.2 line bound= yes
13  .2 .9 line bound= no
14  .4 .8 ellip= .2 .1
15  .5 .6 circ= -.25
16  .4 .4 circ= -.25
17  .2 .3 ellip= .2 .1
18  fini line
19  region 2
20  mu= 1 epsilon= 1 sigma= 0
21  start 0 0
22  .2 0 line bound= yes
23  .2 .3 line bound= no
24  .4 .4 ellip= -.2 -.1
25  .5 .6 circ= .25
26  .4 .8
27  .2 .9 ellip= -.2 -.1
28  .2 1.2 line
29  0 1.2 bound= yes
30  fini
31  region 2 hole 1
32  start .2 .45
33  .35 .45 line
34  .35 .75
35  .2 .75
36  fini
37  region 3
38  mu= 100 epsilon= 1 sigma= 0
39  start .2 .45
40  .2 .75 line bound= no
```

```
41      .3 .75 line
42      .3 .45 line
43      fini line
44      region 4
45      mu= 1 epsilon= 1 sigma= 1e7
46      start .3 .45
47      .3 .75 line
48      .35 .75 line
49      .35 .45 line
50      fini line
51      end
```

Line 1	contains the title we chose for the example problem.
Line 2	defines the grid size and pipe radius.
Lines 3..6	are comment lines.
Lines 7..18	describes perfectly conducting Region 1. Lines 10 - 18 each include explicit declaration of the polygon segment connection type.
Lines 19..36	describe vacuum Region 2, including one hole which contains two additional regions, 3 and 4. Lines 31 - 36 describe the hole. Line 26 assumes **circ** segment type declared in line 25. Likewise, lines 29 and 30 assume **line** type declared in line 28; and, lines 34 - 36 assume **line** type from 33. Note: **bound=** yes is in effect on line 29 and 30.
Lines 37..43	describe a ferrite material in Region 3.
Line 44..50	describe perfectly conducting Region 4.
Line 51	is the end-of-file statement

4.4 Xwake 1.0 Rules And Tips

Physical Considerations:

- Xwake 1.0 applies an open boundary condition valid only for TEM waves at each beam pipe exit. For this reason the length of the beam pipe at both ends of the structure should be selected such that only TEM like waves are reaching the boundary. For most cases a length greater than the maximum radius of the cavity is adequate.

- Xwake 1.0 requires that the entrance and exit beam pipe radius be equal.

- Xwake 1.0 requires that any structure modeled must not extend below the beam pipe radius.

- The step sizes in the r and z directions do not have to be equal as long as, for stability reasons, the ratio of dr/dz is less than 1.5.

- Xwake 1.0 has an intracell resolution of 0.01 times the radial discretization size in the radial direction and 0.01 times the axial discretization size in the axial direction.

- It is recommended that the grid discretization be smaller than 0.1 times the wavelength of the highest frequency component that is of interest. Frequencies higher than that suffer from numerical dispersion. Ultimately the velocity of propagation falls to zero for wavelengths which are twice the grid discretization size. Note that wavelengths become shorter in materials.

- Occasionally the contour mesh generator will not be able to grid a problem. This is usually caused by a problem description file that has an intricate geometry that is much smaller than a grid dimension; making the grid finer will alleviate this condition.

- When run with a contour approximation, Xwake 1.0 is capable of modeling vacuum and metallic materials.

- When run with a stepped approximation, Xwake 1.0 is capable of modeling materials of arbitrary composition; any physically realistic values for permittivity, permeability and conductivity can be assigned to a region.

Problem Description File Syntax:

- The outer grid boundaries of any given problem is always of rectangular shape and, therefore, the parameter "**bound=** yes" is incompatible with "**circ=**" and "**ellip=**" .

- The sequence of parameters in a given line is not important as long as the point coordinates always appear first on the line.

- All coordinates and grid dimensions must be positive.

- A comment can be added anywhere in the file—after the header lines—as long as the first character appearing on the line is a # sign.

- The code is case sensitive and only lower case characters are accepted.

4.5 Glossary

The following is an alphabetical listing of all terms associated with the description file.

Notes:
1) Equal signs are treated as an integral part of the parameter and must be followed by some value. Between the "=" and the value should be a space or tab.
2) When more than one parameter is included on a line, a space or tab must separate the numerical value of one parameter and the next parameter.
3) A "strictly positive" number is greater than zero.
4) "CW" means clockwise and "CCW" means counter clockwise.
5) Real numbers can be represented by:
   a. an optional plus or minus sign,
   b. a sequence of decimal digits possibly containing a single decimal point, and
   c. an optional exponent part, consisting of the letter e or E, and optional sign, and a sequence of decimal digits.
6) All values are in MKS units.

Enables a line to be commented when it appears as the first character on a line.

bound= Declares whether the present connection lies on the problem boundary or not. It is valid with a **line** connection type only. When it is followed by "yes" it indicates that the **line** segment lies on the outer boundary—"no" indicates that the segment does not.

circ= Connects the present point to the previous one with a circular arc. It is followed by a non-zero real number value defining the radius. A positive value indicates a CW connection, and a negative value indicates CCW. A single circular arc segment must not span more than 90^{o}. A full circle must be, therefore, described by four arc segments.

dr= Defines the mesh step size in the r-direction. It appears on the second line in the file. Valid values are strictly positive real numbers.

dz= Defines the mesh step size in the z-direction. It appears on the second line in the file. Valid values are strictly positive real numbers.

ellip= Connects the present point to the previous one with an elliptical arc. It appears in line after a pair of point coordinates. It is followed by two non-zero real number values defining the semi-axis lengths in the r and the z directions, respectively. Only horizontal and vertically oriented ellipses are allowed. Positive values indicates a CW connection, and negative values indicate CCW. Both values must have the same sign. As with a circular arc, a single elliptical arc segment must not span more than 90^{o}.

end Appears by itself on the last line in the file.

epsilon= Relative permittivity. It takes a strictly positive real value. It appears on the second line in a region description.

fini Appears as the last line in every region description and hole description and serves to close the polygon. If connection types or the **bound=** parameter need to be changed, they should follow the word, **fini**.

hole Identifies an area within a region containing a different media characteristic. There can be more that one hole in a region. A hole description is part of the region in which it appears, and is described after the region is described. There can be more than one material contained within a hole. (Materials contained within holes are described as independent regions.) Refer to Figure 3 for the syntax of a hole description. Holes in a region are assigned strictly positive integer numbers and increase sequentially starting with "1".

line Used in region and hole descriptions indicating that a straight line segment connects the present point to the previous one.

mu= Relative permeability. It is followed by a strictly positive real number value. It appears on the second line in a region description.

region Refers to an area containing a homogeneous material. Each region in the structure must be described. Regions cannot overlap. All regions are assigned strictly positive integer numbers and increase sequentially starting with "1". Refer to Figure 2 for the syntax of region descriptions.

rmax= Defines the maximum grid dimension in the z direction. Values are strictly positive real numbers. It appears on the second line in the file.

rmin= Defines the minimum grid dimension in the z direction. Currently, this value can only be 0. It appears on the second line in the file.

rpipe= Defines the beam pipe radius. It takes a strictly positive real value.

sigma= Conductivity. It takes a positive real value. It appears on the second line in a region description.

zmin= Defines the minimum grid dimension in the z direction. Currently, this value can only be 0. It appears on the second line in the file.

zmax= Defines the maximum grid in the z direction. Values are strictly positive real numbers. It appears on the second line in the file.

5.0 INSTALLATION

Xwake is available on the Internet via anonymous ftp at `calvin.fnal.gov`. Executables and their associated resource files are found in sub directories whose names match various machine types. In each directory are an executable, `xwake`, a Motif resource file, `XWake`, and a readme file. Documents including this one can be found in the `/doc` sub directory.

When installing `xwake` on a machine for general availability, Motif convention specifies saving the resource file in the directory `/usr/lib/X11/app-defaults`. The executable will go wherever your system administrator chooses. If you run this program with the executable in your own area, save the resource file in your root directory. Motif looks, first, in the `/app-defaults` directory and, next, in your root directory for a resource file specific for an application.