Appendix B

Using Prism with CMAX

You can use Prism with source code you have translated from FORTRAN 77 to CM Fortran using the CMAX Converter. Prism lets you:

- View both the FORTRAN 77 source code and the corresponding CM Fortran source code at the same time, using Prism's split source window. Or, you can view either code individually.
- Set breakpoints and create other events in terms of either the FORTRAN 77 or CM Fortran code.
- View the call stack in terms of either the FORTRAN 77 or CM Fortran code.
- View performance data in terms of either FORTRAN 77 or CM Fortran code.

This appendix describes how to use Prism with a CMAX-translated program.

B.1 How Prism Can Display Both Source Files

To display both FORTRAN 77 and CM Fortran source files, Prism uses a mapping file that is created as part of the CMAX translation. This mapping file is named *filename* `.ttab`, where *filename* is the name (minus the extension) of the FORTRAN 77 and CM Fortran source files. If Prism can't find this file, it won't be able to load the FORTRAN 77 source file.

Copyright © 1991–1994 Thinking Machines Corporation

B.2 Splitting the Source Window

Begin by loading the executable CM Fortran program into Prism.

You then split the source window to display the corresponding FORTRAN 77 code by following these steps:

1. Right-click in the source window to display the source-window popup menu.
2. Choose **Show source pane** from this menu.
3. This displays another menu. Choose **Show .f source** from this menu.

You then see a split screen like that shown in Figure 48. The CM Fortran source code is in the top pane; the corresponding FORTRAN 77 source code is in the bottom pane.

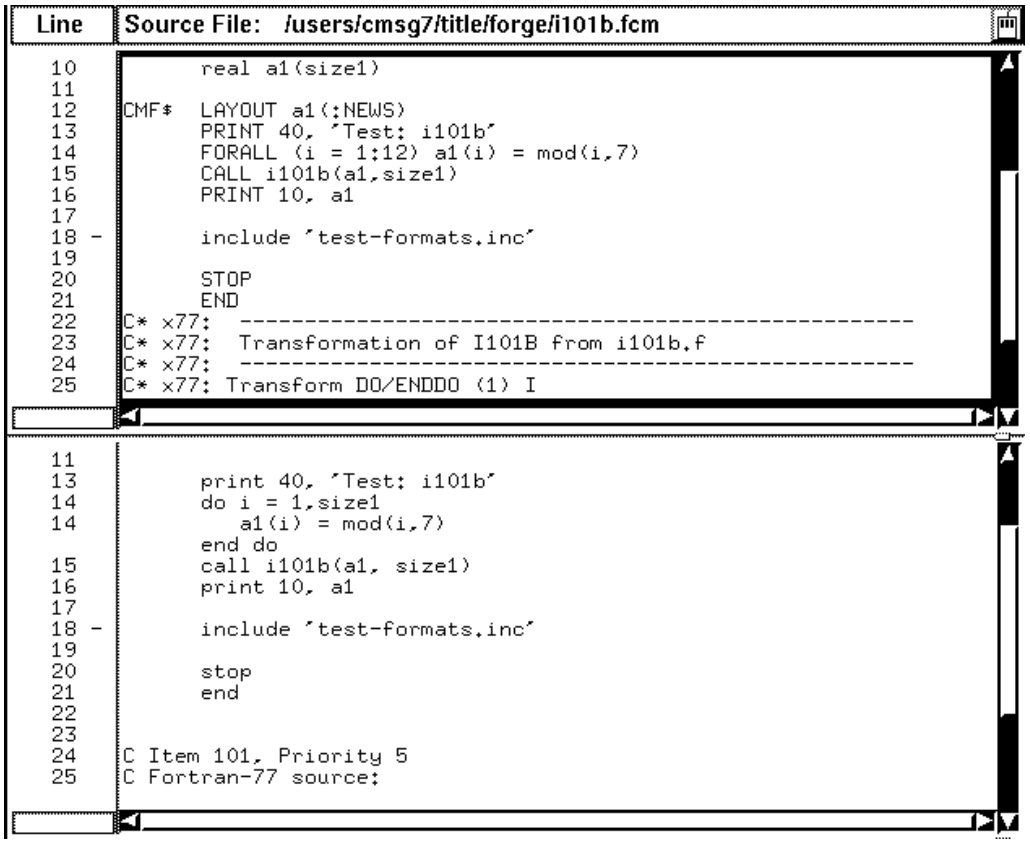

Figure 48. CM Fortran and FORTRAN 77 code in a split screen.

Version 2.0, April 1994
Copyright © 1991–1994 Thinking Machines Corporation

When your screen is split, breakpoints you set in the line-number region of one pane also appear at the corresponding line in the other pane.

To return to a single source window, put the mouse pointer in the pane you want to delete, right-click, and choose **Hide this source pane** from the popup menu.

NOTE: You can also choose **Show .s source** from the **Show source pane** menu; this displays the assembly code for the executable program. You can therefore have a three-way split of your source window, displaying the CM Fortran, FORTRAN 77, and assembly code versions of the program.

B.2.1 From the Command Line

To split the source window, issue the **`show`** command, using as an argument the file extension of the source file you want to see in the other pane. For example, when you first load the program, only the CM Fortran source code is visible. To see the FORTRAN 77 source code in a separate pane, issue the command:

```
show .f
```

To return to a single source window, issue the **`hide`** command, specifying the file extension of the source code you no longer want to see. For example, to display only the FORTRAN 77 code, issue the command:

```
hide .fcm
```


Copyright © 1991–1994 Thinking Machines Corporation

B.3 Using the Master Pane

When you split the source window, the top (CM Fortran) pane is highlighted; this is the *master pane*. Left- or middle-click in the other pane to make it the master.

The master pane controls several aspects of the way Prism operates when the screen is split. Specifically:

- Scrolling through the master pane causes the slave pane to scroll to the corresponding location. You can scroll the slave pane independently, but this does not cause the master pane to scroll.
- The line numbers shown in the slave pane refer to the source code in the master pane. Thus, several FORTRAN 77 lines in the slave pane could have the same line number, if they were all converted to a single CM Fortran statement. Similarly, if the FORTRAN 77 pane is master, the CM Fortran pane may skip a line number, if it doesn't have code corresponding to that FORTRAN 77 line.
- Prism interprets all unqualified line numbers in commands as referring to the source code in the master pane. You can still refer to a line number in the other source code, but you must qualify it with the filename. For example, if CM Fortran is the master pane, you would specify a breakpoint in the FORTRAN 77 code like this:

  ```
  stop at "foo.f":20
  ```

- Prism displays line numbers for the master source code in the event table, the **Where** window, and in messages in the command window.
- Prism displays the files for the master source code in the **File** window.
- Prism displays master source code in response to a `list` command.
- Source-line histograms in the **Performance Data** window show the master source code.
- The `?` and `/` search commands search in the master pane only.
- If you choose **Edit** from the **Utilities** menu, the master source code appears in the editor.

Version 2.0, April 1994
Copyright © 1991–1994 Thinking Machines Corporation

B.3.1 From the Command Line

To choose the master pane from the command line, issue the **select** command, specifying the extension of the source code you want to be in the master pane. For example, to make the FORTRAN 77 code be in the master pane, issue this command:

```
select .f
```

B.4 Displaying Corresponding Source Lines

Prism lets you graphically display the line or lines in one pane that correspond to a line in another pane. Press the **Control** key and left-click in the line-number region next to the line you are interested in. A pound sign (**#**) appears next to the line; the same character appears next to the corresponding line(s) in the other pane.

B.5 Debugging

You can debug in terms of either the FORTRAN 77 or CM Fortran source code. For example, setting a breakpoint in one source code causes it to be set in the corresponding location in the other source code.

If you hide the CM Fortran pane, it will look as if you are debugging your FORTRAN 77 code directly. Note, however, that this is not exactly the case — you are actually debugging a CM Fortran executable program. This means that you will not be able to set a breakpoint at a line that has been optimized away in the translation. For example, if you try to set a breakpoint in the middle of a **DO** loop, Prism insists on putting the breakpoint at the beginning of the loop. Similarly, you can't print a FORTRAN 77 variable that has been optimized away.

Copyright © 1991–1994 Thinking Machines Corporation

B.6 Analyzing Performance

You can use Prism to analyze the performance of your translated CM Fortran program. This will tell you if CMAX didn't translate a FORTRAN 77 construct into the most efficient CM Fortran equivalent. For example:

- Unvectorized loops are treated as serial code in CM Fortran and are executed on the partition manager; if you use Prism's performance analysis feature, these loops will show up as using the **PM CPU (user)** resource (on the CM-5) or the **FE CPU (user)** resource (on the CM-2/200).
- FORTRAN 77 code that could be translated into a **`CSHIFT`** may be translated as an assignment of array sections. This will generate unnecessary Send/Get communication.

Note that you have two choices in working with your program:

- You can edit the FORTRAN 77 code, and then use CMAX to retranslate the code into CM Fortran.
- You can edit the translated CM Fortran code directly. If you do this, you can no longer use Prism's split source window to view the original FORTRAN 77 source, since the FORTRAN 77 code and the CM Fortran code won't be equivalent.

B.7 Using Commands-Only Prism

Although you don't have the benefit of using the split source window, you can use commands-only Prism with a CMAX-translated program. Use the **`select`** command to specify the version of the source code in which you want to debug. For example, issue this command to debug in terms of the FORTRAN 77 source code:

```
(prism) select .f
```

To subsequently debug in terms of the CM Fortran source code, issue this command:

```
(prism) select .fcm
```


Version 2.0, April 1994
Copyright © 1991–1994 Thinking Machines Corporation