

# AWIPS/LINUX

(Advanced Weather Interactive Processing System on Linux)

U. Herb Grote

NOAA Forecast Systems Laboratory, Boulder, CO

## 1. INTRODUCTION

The Forecast Systems Laboratory has been working closely with the National Weather Service for many years to develop the Advanced Weather Interactive Processing System (AWIPS). All weather forecast offices in the United States are currently using AWIPS in their day-to-day operation. The latest software release (Version 4.3) contains over two million lines of code, of which approximately 500,000 are part of the FSL-developed D2D display code.

The initial objective of the AWIPS/Linux project is to convert the existing AWIPS code to run on a Linux platform. The longer term objective is to develop a new AWIPS system architecture around the Linux operating system that incorporates the latest advances in sensor technology and forecast techniques.

## 2. THE AWIPS SYSTEM

The AWIPS software provides a complete set of capabilities for use in a NWS forecast office. It performs real-time data acquisition, data management, application processing, and display generation. Forecasters have a diverse set of meteorological data at their disposal to help them produce timely and accurate weather forecasts. These data include a selection of forecast models, radar and satellite imagery, surface and upper air observations, and observations from hydrological, buoy, and profiler systems.

At a high level, the AWIPS system architecture (Figure 1) can be partitioned into four major components: communications processing, application processing, data storage and retrieval, and display manipulation. Each component has one or more computers that are interconnected by a local area network. National guidance products arrive at the communications processor, handling the communications protocol and some data filtering, and are forwarded to the applications server for decoding and storage formatting. The products are then stored on the disks attached to the data server and remain there until they are removed by a data purger. When a forecaster at the workstation requests a product for display, the data are retrieved from the data server and processed for display on the workstation.

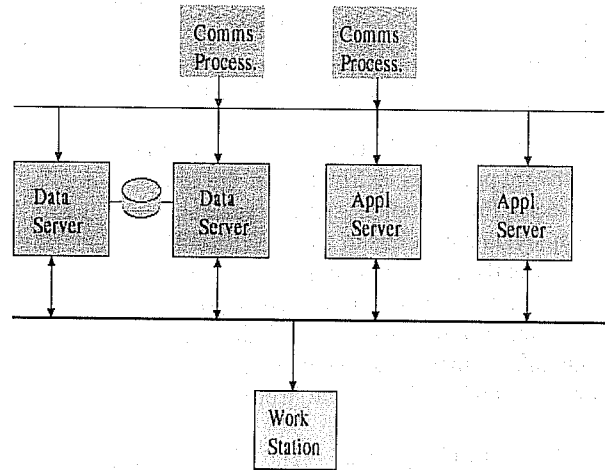


Figure 1 AWIPS Block Diagram

## 3. LINUX CONSIDERATIONS

The rapid advances in computer technology have spurred FSL's interest in providing a full function meteorological workstation on a low-cost, high-performance processing platform. FSL considered several candidates (including WindowsNT) for the operating system, but Linux became the top candidate because of its capability and increasing popularity.

### 3.1 Strengths

Although Linux has many strengths, the five characteristics listed below are of particular interest:

- *Low Price* – The costs of similarly equipped Hewlett Packard (HP) hardware and a PC were obtained toward the end of 1999 (Figure 2). Although these costs may have changed somewhat since then, we still believe that there is a potential cost advantage in using a PC running Linux.
- *Stability* – The Linux system appears to be very stable. FSL has been running a significant number of Linux PCs for development and testing for an extended period of time without encountering any problems. Similarly, an informal survey of users outside FSL seemed to confirm that Linux is as stable, if not more so, than WindowsNT.

Informix 10 User License Server and Development	\$1,000	Informix 10 User License Server and Development	\$35,000
Desktop PC	\$3,000	New HP C200 (Refurbished)	\$26,000 (\$14,000)
LINUX OS	\$0	HP - UX	(Included)
TOTAL	\$4,000	TOTAL	\$61,000

Figure 2 Cost Comparison (1999)

- *Performance* – FSL has ported three major software packages to Linux PCs: the Interactive Forecast Preparation System Graphical Editor, 3-D Display, and portions of the AWIPS 2-D Display. In all cases the performance is very good and the performance per cost ratio is considerably better than on other systems. As expected, there are some capabilities that do not execute as fast on the PC as on some high-end hardware from other vendors.

- *Standard Compliance* – The Linux operating system is POSIX-compliant and its components support all relevant ANSI and ISO standards.

- *Diverse HW Support* – Although FSL is planning to initially run AWIPS/Linux on Intel processors, the knowledge that Linux is supported on 68-k, SPARC, ALPHA, Power PC, PA-RISC, and other computer chip sets favors Linux as the operating system.

### 3.2 Perceived Weaknesses

As is true with most other evolving technologies, the many strengths of Linux are accompanied by some perceived weaknesses, as follows.

- *User Unfriendly* – With many years of experience using various versions of Unix, FSL staff did not perceive Linux to be a problem. FSL developers are evaluating various graphical user interfaces (GUI) available for Linux, such as KDE, Gnome, and fwm. Their experience with these interfaces will allow them to recommend the preferred GUI for use in AWIPS.

- *Installation Complexity* – The Linux operating system is available at no cost from sites on the Internet, or it can be obtained at a small cost from a number of different distributors. The Linux distributors tend to package the operating system with drivers and desktop environments, and make it available on a compact disk. FSL is using the latest Red Hat distribution of Linux. In most cases it took less than an hour to install the operating system, and the installation was relatively easy. Some difficulties can arise with certain peripherals and graphic cards; however, FSL found that

these could usually be resolved readily with a search on the Internet. Laptops can be a little more challenging since they tend to use nonstandard graphics cards. Again, a Web site provides considerable guidance for installing Linux on a laptop (University of Texas, 2000). For the most part, we have found the installation to be fairly easy. Moreover, some PC vendors (such as Dell Corporation) will preinstall Linux on the system at the request of the purchaser.

- *Poor Documentation* – There is a large amount of documentation available for Linux on the Web. However, it can require some effort to locate the desired documentation, which may not always be uniform in depth and presentation. A Linux Documentation Project has been formed to help provide a central access point to Linux documentation. In addition, an activity is under way to try to provide more uniform documentation for Linux.

- *Security Concerns* – The Linux operating system is a true Unix operating system and thus provides the same level of security as other Unix operating systems. A frequent concern is that since the source code is readily available, it will be easy for a “hacker” to find ways to penetrate the system. AWIPS has its own secure network which should make it difficult for anyone outside the network to gain access to the systems. Also, Linux users usually find that when a security problem is discovered, a fix is available in a matter of hours or, at most, days.

- *Lack of Support* – Support for Linux is available from hardware vendors, such as Hewlett Packard, IBM (in progress), and an organization called LinuxCare. Many users have found that support from the Linux community is quite adequate. Although FSL has not needed to request help from the user community on specific Linux problems, we requested help on a g77 compiler bug, which was corrected within a few hours.

- *Uncertain Roadmap* – The direction of Linux is determined by industry trends in hardware and software technology, not by corporate decision. For example, as 64-bit processor and multiple processor architectures are being introduced, the Linux community is quick to ensure that Linux will be able to take advantage of these technological advances. Thus, the user community determines which direction Unix development evolves.

### 3.3 Groundswell Support

According to IDC Research (IDC, 1999), Linux is the fastest growing server-environment operating system,

with about 17% of all the servers using Linux. The popularity and rapid growth of Linux servers and workstations during the last few years are illustrated in Figure 3.

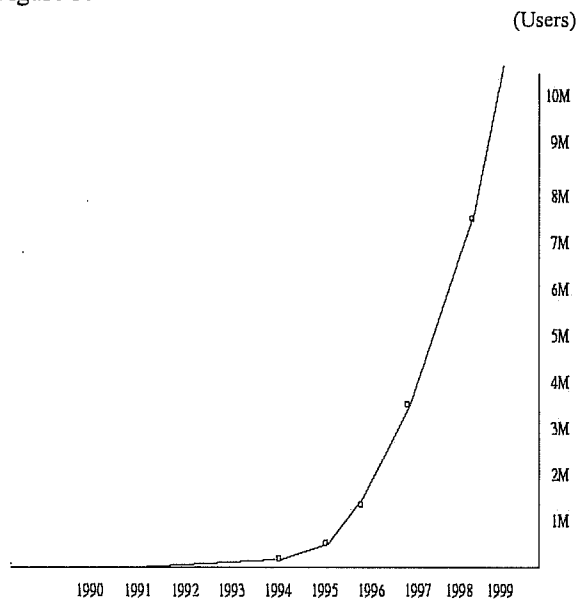


Figure 3. Linux Growth

### 3.4 Future Trends

Linux continues to evolve and take advantage of advances in technology. Considerable effort is being invested in supporting 64-bit processors and providing fast performance for multiple processor architectures (Symmetric Multi-Processing). There is also an increasing number of Beowulf clusters, a cluster of processors running Linux that challenge current high-performance computers. A good discussion of the strengths and weaknesses of Linux can be found on the Web (Open System Opinion, 1999)

## 4. LINUX PORTING EXPERIENCES

The issues faced in porting software from another operating system to Linux varies greatly, depending on the type and complexity of the software. A discussion follows of some of the issues faced by FSL in porting the AWIPS software to a Linux platform.

### 4.1 Graphics

The AWIPS display software uses the X11 libraries and their implementation for specifying the color model for the X-window by using the X-visual. A somewhat unique feature of the HP hardware and software is the ability to specify multiple concurrent X-visuals. The AWIPS display contains five image display windows, of which four use the 24-bit visual and the fifth uses an

8-bit visual. In addition to multiple X-visuals, the HP graphic card provides several overlay and underlay planes.

The lack of these features on the PC graphic cards required that the display be set to the 24-bit color mode. This placed a significant extra memory and processing burden on the machine, since pixmaps (containing the image to be displayed in a window) increased by a factor of four (from 8 to 32 bits/byte). The simple operation of removing a graphic overlay from an image required that the entire image be re-rendered instead of merely turning off a graphic plane. Also, functions such as changing the image color required re-computing each pixel value instead of modifying a color lookup table.

### 4.2 Programming Languages

The AWIPS software is written in C, C++, Fortran, and scripting languages such as Tcl/tk. It is compiled on the HP hardware using HP's CC and Fortran compilers. A requirement in porting the software to a PC was to ensure that the modified code would still run on the HP hardware with the HP compilers. As a result, FSL compiles the code daily on the HP using the GNU and HP compilers, and on the PC using the GNU compilers.

The compilers for the AWIPS/Linux system are the GNU gcc and g77. Since these compilers are more ANSI compliant than the HP CC (c-front) and Fortran compilers, some of the code needed to be modified to make it ANSI compliant. For example, the GNU g77 compiler does not support records and uses much stronger data typing than the HP compiler. Some of FSL's legacy Fortran code frequently used integers in Boolean expressions. Other differences in Fortran compilers were in the intrinsic functions, which were dealt with by writing a "g77 wrapper" for the Fortran functions.

### 4.3 Processor Architecture

The ordering of bytes within a word for a PC is different than for the HP. As a result, data files that were generated on the HP were read incorrectly on the PC. Since the code has to be executable on the PC and the HP, a run-time test is performed to determine if the bytes need to be reordered. Certain data files (netCDF and shapefile) use a data representation that makes them read correctly on both types of machines. The software that reads binary cardiographic (map backgrounds) data and certain tables was affected the most by the difference in byte order. So far, FSL has not fully investigated the impact of the new 64-bit processors on the software.

Another challenge involved the difference in the way processing threads are implemented by different hardware and software. The HP systems used DCE threads versus POSIX threads for Linux. A crucial component of the AWIPS system is the Interprocess Communications (IPC). Tens of thousands of messages are exchanged every day between processes. The IPC software was designed to use multiple threads to efficiently and reliably handle large volumes of messages; the IPC software had to be rewritten for Linux.

#### 4.4 Development Tools

FSL generally found the development tools available for Linux to be adequate, including compilers, debuggers, performance analyzers, and editors. However, FSL was unable to locate a good software tool (such as the HP Purify software) for finding program memory leaks.

Most AWIPS commercial off-the-shelf (COTS), as well as public domain software, is available for Linux. The COTS Informix relational database manager is a key component of the AWIPS system. COTS accounts for a measurable portion of AWIPS code.

### 5. PROJECT STATUS

The AWIPS software is rapidly being ported to Linux. The first step of this effort, porting all of the software to the gcc compiler, is complete. The second step, moving the software from the HP to a PC running Linux, is partially complete. In addition, the software infrastructure and the entire product display software to Linux has been ported. The system was successfully demonstrated at the 16<sup>th</sup> IIPS Conference Exhibit of the Annual Meeting of the American Meteorological Society in January 2000. Concurrent with the effort to port the software, FSL is investigating and assessing new system architectures for the AWIPS system, which include PCs and Linux as a key component.

#### References:

University of Texas, *Linux in Laptops*, January 2000  
<http://www.cs.utexas.edu/users/kharker/linux-laptop>.

IDC, *Linux Operating System Market Overview*, March 1999 <http://idc.com/> and <http://www.cnn.com/TECH/computing/9904/02/linuxgrow.ent.idg>.

Ganesh Prasad, *The Practical Manager's Guide to LINUX*, 1999;  
<http://www.osopinion.com/Opinions/GaneshCPrasad/main.html>.