1996 NASA/ASEE SUMMER FACULTY FELLOWSHIP PROGRAM
JOHN F. KENNEDY SPACE CENTER
UNIVERSITY OF CENTRAL FLORIDA

VISUALIZATION OF REAL-TIME DATA

Dr. Ryan Stansifer, Associate Professor
Computer Science Department
Florida Institute of Technology
Melbourne, Florida

KSC Colleague - Peter Engrand
Artificial Intelligence/Expert Systems

Contract Number NASA-NGT10-52605

August 9, 1996

ABSTRACT

In this project we explored various approaches to presenting real-time data from the numerous systems monitored on the shuttle to computer users. We examined the approach that several projects at the Kennedy Space Center (KSC) used to accomplish this.

We undertook to build a prototype system to demonstrate that the Internet and the Java programming language could be used to present the real-time data conveniently. Several Java programs were developed that presented real-time data in different forms including one form that emulated the display screens of the PC GOAL system, this system is familiar to many at KSC. Also, we developed several communicating programs to supply the data continuously. Furthermore, a framework was created using the World Wide Web (WWW) to organize the collection and presentation of the real-time data.

We believe our demonstration project shows the great flexibility of the approach. We had no particular use for the data in mind, instead we wanted the most general and the least complex framework possible. People who wish to view data need only know how to use a WWW browser and the address (the URL). People wanting to build WWW documents containing real-time data need only know the values of a few parameters—they do not need to program in Java or any other language. These are stunning advantages over more monolithic systems.

1 Introduction

1.1 Real-Time Data

During the operation of the shuttle, sensors are monitoring many of the shuttle systems. This data goes to special hardware at the Launch Control Center (LCC) known as the Common Data Buffer (CDBF). Lots of data is collected, approximately 30,000 measurements. These measurements are continually changing—some of them can change rapidly at certain times. The data is used in monitoring the operation of the shuttle and in analyzing substems for saftey, performance, technolgical improvements, etc.

The goal of this project to design a new information system structure to meet all these needs.

2 Related projects

Several systems use the real-time data from the CDBF. These systems vary vastly in purpose and sophistication.

2.1 PC GOAL

The goals of the PC GOAL system presents shuttle data on schematic like screens described by character-oriented DSP files. Figure 1 shows one of the PC GOAL display screens. The system PC GOAL system requires Intel computer and special networking hardware and software to work. The data collection engine is fast and reliable.

2.2 PAT

Rockwell's propulsion advisor tool (PAT) is designed to do intelligence analysis. It is not this aspect that is relevant here at the moment, but rather its the overall architecture. PAT is an X Window System application. This provides excellent and efficient graphics.

The X Window System requires that the software be ported and distributed to all the platforms. Also the computer users would have to be running the X Window System. Actually more different platforms run the X Window System than support Java, but currently more users are familiar with WWW browsers than with the X Window System.

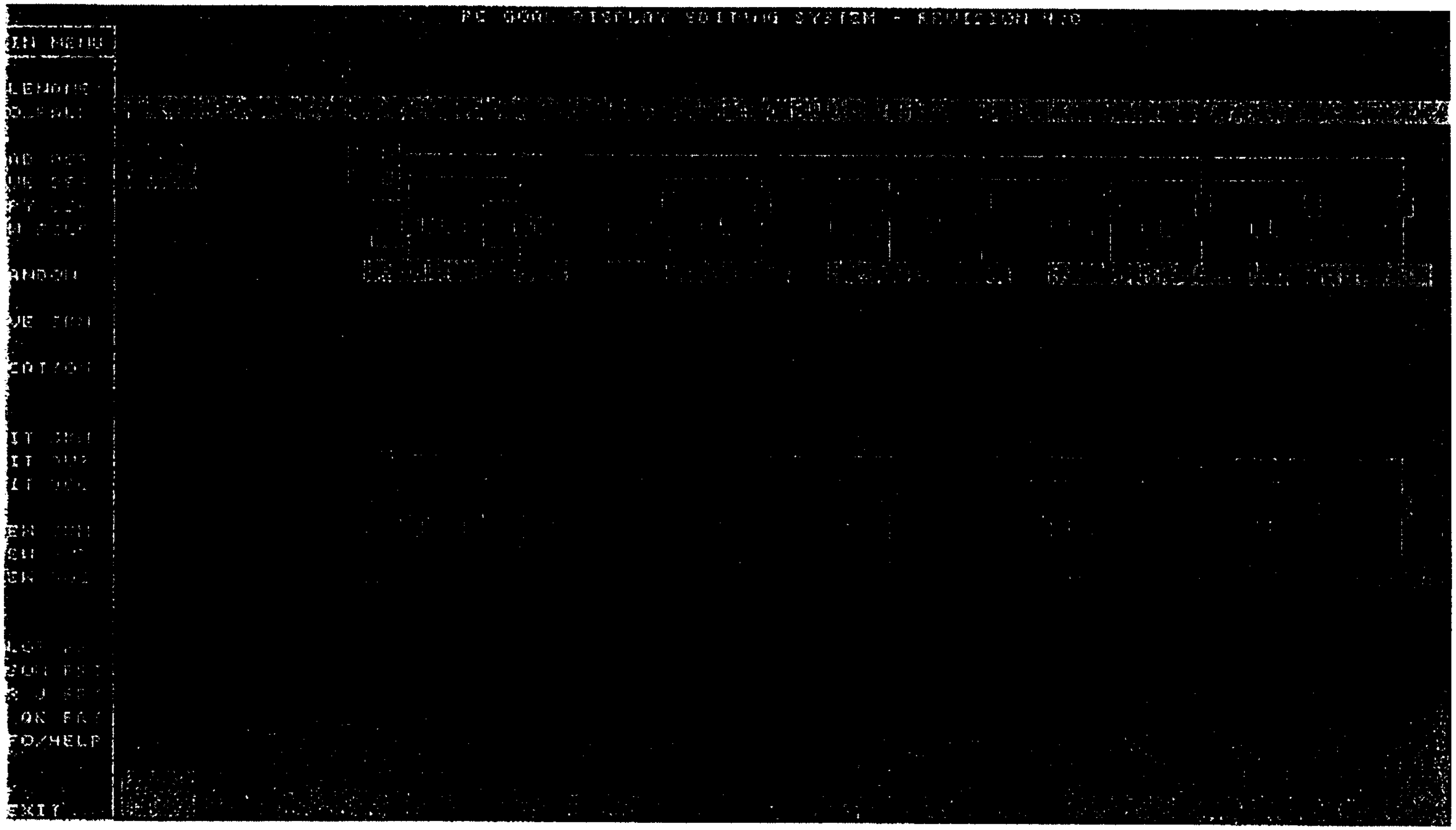


Figure 1: One of PC GOAL display screen

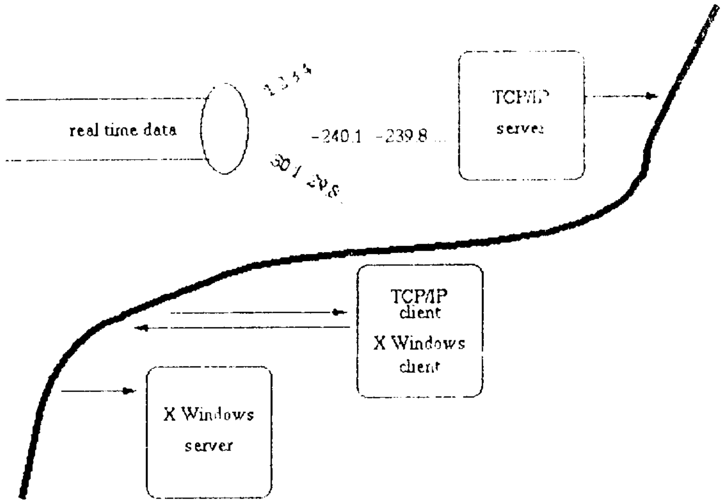


Figure 2: Real-time data reaching an X Window System application

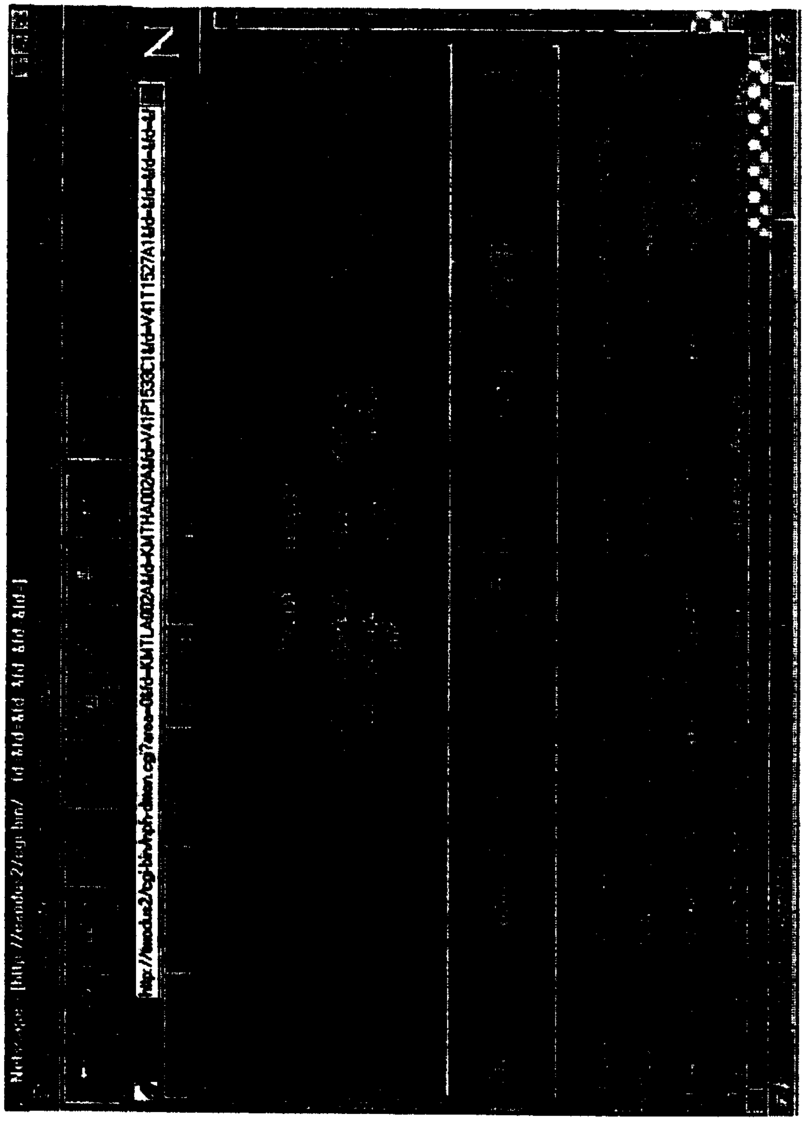
2.3 Exodus

The Exodus project displays its data on WWW documents using CGI programs. Figure 3 shows one the documents produced by the CGI program with some real-time data. (For some reason, unreasonable numbers and strange characters appear in the place of the good data, but that is not the point here.) Periodically another document is produced with the data on it and sent to the WWW browser. This requires a lot of redundant network traffic when on a few numbers have changed. Also graphics are hard to produce on an HTML document without resorting to graphic images that would have to be transmitted each time to the browser.

3 Java applets

The key difference with using Java is that the applet makes a connection to a data server and only the data is transmited across the network after the initial class is transmited. Figures 4 and 5 depict the situation.

The advantages are in using Java are even greater than narrow technical merits suggest. Administratively, the operation of real-time data service using Java is much better. The Java applet is written once and executed remotely; no porting has to be done. Also, the



Netscape - [http://exodus2/cgi-bin/ fd &fd=&fd &fd &fd &fd=]

http://exodus2/cgi-bin/hph-dmon.cgi?area=0&fd=KMTLA002A&fd=KMTHA002A&fd=V41P1533C1&fd=V41T1527A1&fd=&fd=&fd=&fd=&f

Figure 3: Four FD's monitored by Exodus DMON

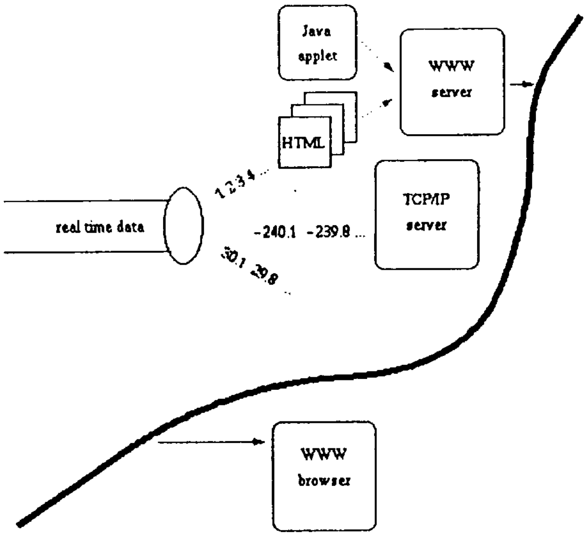


Figure 4: A WWW server distributes a Java applet

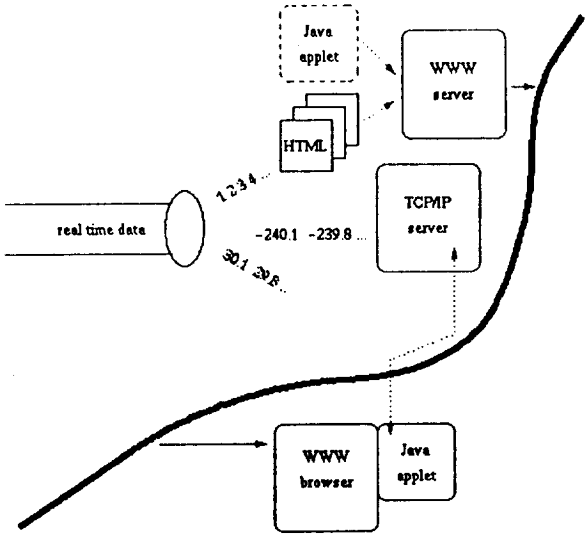


Figure 5: Then, the applet connects with the TCP/IP data server

latest version of the applet is always distributed to the user; there is no version control problem. Finally, the operation of the service is easy as browsers are ubiquitous; no training is required.

4 Using the Display Applet

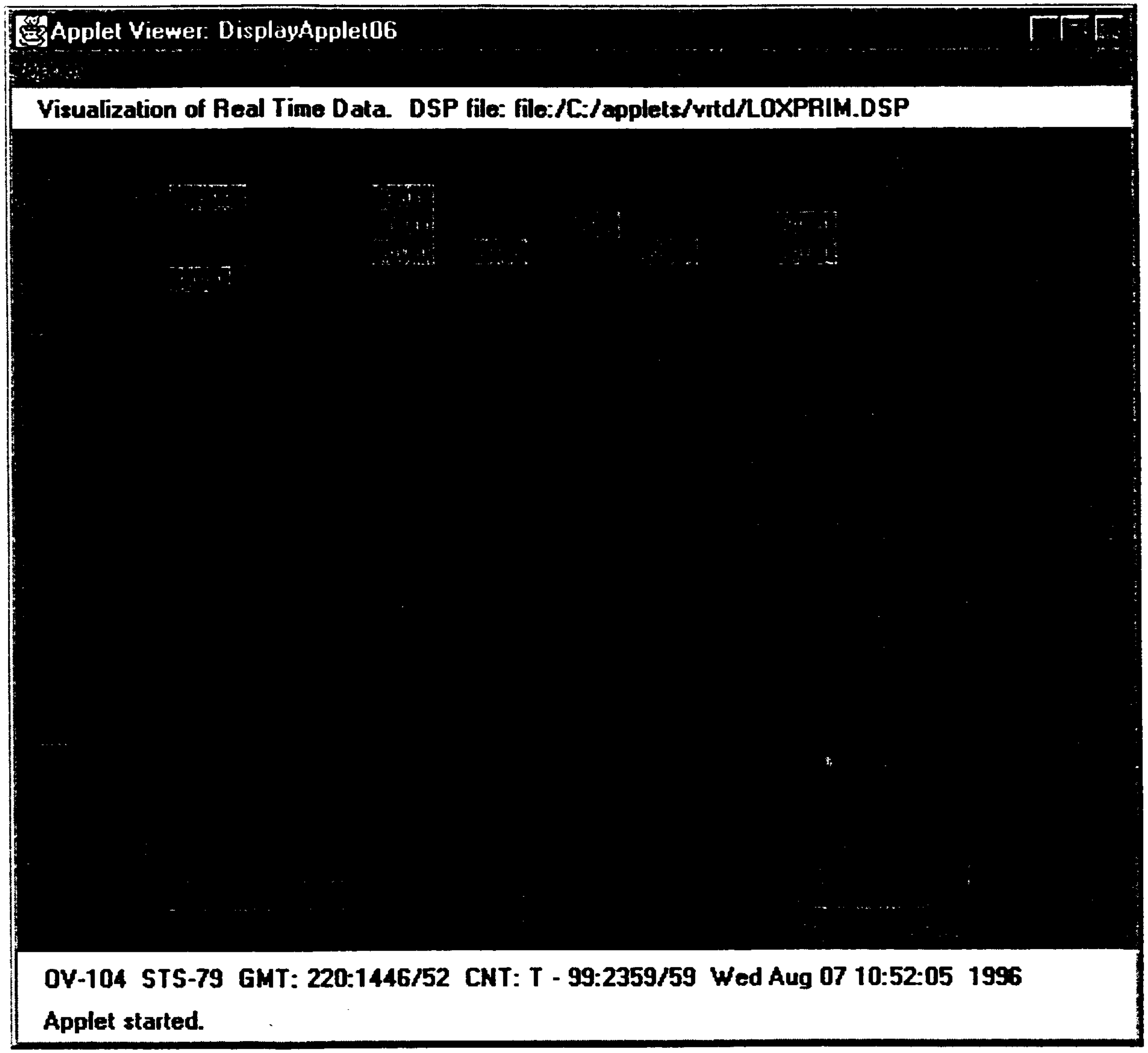
Adding real-time data to an HTML document is easy in the framework we have developed. Here is an an example of using the display applet. The indicated FD's are displayed on the PC GOAL display background LOXPRIM. The numbers are used into place the data in a rectangle region of the image. The format governs how the data is displayed.

```
<applet code="DisplayApplet06" width=576 height=465>
   <param name="server"     value="ecto1">
   <param name="orbiter"    value="104">
   <param name="background" value="LOXPRIM.DSP">

   <param name="0FD"     value="GLOP0006A">
   <param name="0class"  value="TextDisplayClass">
   <param name="0format" value="%5.1f">
   <param name="0pos"    value="80,28,40,14">

   <param name="1FD"     value="GLOQ0229A">
   <param name="1class"  value="TextDisplayClass">
   <param name="1format" value="%4.0f">
   <param name="1pos"    value="184,28,32,14">
</applet>
```

5 Conclusions and Future Work

We have completed a prototype implementation in the Java programming language in which real-time data can be display on top of images, in PC GOAL DSP files, or in individual components. Figure 6 shows a Java applet emulating the PC GOAL screen LOXPRIM.

The graphical user interface is primitive, but until demand requires a particular function, making the interface more elaborate would be less productive. The work now needs to turn to the client/server architecture and design of appropriate network protocols and communication programs.



Visualization of Real Time Data. DSP file: file:/C:/applets/vrtd/LOXPRIM.DSP

OV-104 STS-79 GMT: 220:1446/52 CNT: T - 99:2359/59 Wed Aug 07 10:52:05 1996

Applet started.

Figure 6: Real-Time Data in an Emulated PC GOAL screen

6 Acknowledgments

The impetus of this work is due to Peter Engrand (NASA) and Charlie Goodrich (I-Net). The rest of the Advanced Software group, in particular, Steve Beltz and John Dockendorf, contributed greatly in hosting me and providing facilities, The group at Rockwell led by James Engle, in particular Randy Lane and Van Bullington, made the project possible by providing a server for the real-time data. A special thanks is due to Roger Lee (USA) for answering questions about the "scan server." I gratefully acknowledge the contribution of Roger Johnson (UCF) and Greg Buckingham (NASA) who ran the NASA/ASEE Summer Faculty Fellowship Program with efficiency and enthusiasm.

Finally, I wish to thank the summer interns Paul Bryer (Florida Tech) and Kevin Gillette (Princeton). Paul wrote the Java component classes and Kevin wrote the Java code for interpreting DSP files and the code communicating with the "scan server."

A Links

The scientific literature is no particular use in understanding the context or methods of this work, so in lieu of references I have listed a number of links to material on the WWW that provides additional information pertaining to the project.

Books – a list of books on Java

http://www.cs.fit.edu/~ryan/ogi/java-books.html

Demonstration using DSP files – a browser that understands Java can view real-time data by staring at this URL and picking one of the PC GOAL display screens.

http://ecto1.ksc.nasa.gov:8600/ryan/dsp/

Exodus – The Exodus project

LPS http://lpsweb.ksc.nasa.gov/

PC GOAL http://lpswebksc.nasa.gov/SDC/PCGOAL/homepage.html

Personel of the Advanced Systems and Analysis Division

http://www-de.ksc.nasa.gov/de/dm/dm-asd/personnel.html

Ryan Stansifer – author of this report

http://www.cs.fit.edu/~ryan/