BigSignal 2000: Doing Remote Geology in Antarctica from Home

Brian Y. Woo

2000

Advisor: Prof. Khosla

Electrical & Computer ENGINEERING
BigSignal 2000: Doing Geology In Antarctica At School

Brian Woo
5/8/2000
M.S. Final Report
# Table Of Contents

List Of Figures .................................................................................................................. iii  
List Of Appendices ............................................................................................................ iv  
Abstract .............................................................................................................................. v  
Executive Summary ......................................................................................................... vi  
  Status Quo .................................................................................................................... vi  
  2000 Project ................................................................................................................ vi  
  Applet ............................................................................................................................ vi  
  Future .............................................................................................................................. vi  
BigSignal 2000: Doing Geology In Antarctica At School ............................................... 1  
Road To 2000 Project Deployment ................................................................................ 3  
  Nomad .......................................................................................................................... 3  
  Original BigSignal Project ...................................................................................... 3  
  Introduction To The 2000 Project ............................................................................ 4  
BigSignal 2000 Applet .................................................................................................. 6  
  Database ...................................................................................................................... 7  
  Timeline/Daily Reports ............................................................................................ 7  
  Map ............................................................................................................................. 7  
  Rock Comparison Tool ............................................................................................ 7  
  Panoramic Picture Viewer ....................................................................................... 7  
  Integration ................................................................................................................... 7
Problems And Solutions ........................................................................................................ 7

- Dynamic data errors ........................................................................................................ 7
- Memory consumption/applet speed .............................................................................. 7
- Java 1.1 incompatibility .............................................................................................. 7
- Inconsistent browser handling .................................................................................... 7

2000 Project And Beyond .................................................................................................. 7

- 2000 Project Deployment Summary ........................................................................... 7
- Future Projects ............................................................................................................. 7

Conclusions ....................................................................................................................... 7

Works Cited ...................................................................................................................... 7

Appendix A: DatabaseRecord Object ............................................................................ G

Appendix B: Rock Library ............................................................................................... G

Appendix C: Classroom Curriculum ............................................................................... G

Appendix D: Pictures From Deployment ......................................................................... G
List Of Figures

Figure 1: Applet interface .................................................................................................... 6
Figure 2: Timeline/Daily Reports ....................................................................................... 7
Figure 3: Map ...................................................................................................................... 7
Figure 4: Applet utilizing grid for overhead map ............................................................... 7
Figure 5: Rock Comparison Tool ....................................................................................... 7
Figure 6: Panoramic Picture Viewer ................................................................................... 7
Figure 7: Continuous scrolling for panoramic pictures ..................................................... 7
List Of Appendices

Appendix A: DatabaseRecord Object ................................................................. G
Appendix B: Rock Library .................................................................................. G
Appendix C: Classroom Curriculum ................................................................. G
Appendix D: Pictures From Deployment ........................................................... G
Abstract

This report examines the BigSignal 2000 project, a second pilot project for connecting students in middle and high school to the Nomad rover in Antarctica to do remote geology. Topics discussed include the potential of such projects that utilize the Internet to enhance learning, the technology in the form of a Java applet to implement the robot console, and the deployment of the robot and project. The analysis of the applet was done by looking at back-end and front-end components individually, then explaining their integration. Future considerations and projects beyond this one using the same type of technology-enhanced learning are also considered.
Executive Summary

Status Quo
In the American education system, science taught to middle and high school students attempts to involve the students in the subject matter through laboratory experiments. However, many of these experiments fail to engage the students' interests because they are preconceived. Following certain procedure steps leads to observing proper results, but doesn't allow students to experience the decision-making and possibility of new discovery as true science does.

2000 Project
BigSignal 2000 has the unique position of being able to allow students to experience just that. Tied to an actual NASA mission that sends robot rovers to Antarctica to search for meteorites, this project utilizes the interactive, immersive capabilities of the Internet to connect students with a real science mission. Even more than connecting them, this project seeks to allow students to analyze the same data that the robot (in this case, Nomad) does to make their own conclusions and discoveries. The main technology used for this was a Java applet to function as a robot console or mission interface.

Applet
The applet was an all-inclusive learning tool, involving a database, pictures, visual representation of data, and chronological and geographical presentation of the mission. The report analyzes the applet component by component, then describes its integration. The report continues by revealing the testing process and changes that occurred during the mission.

Future
This project was a pilot for EventScope, another multimedia, Internet-enhanced project seeking to immerse and involve students in actual planetary science missions in greater scope, depth, and breadth.
BigSignal 2000: Doing Geology In Antarctica

At School

The American education system desires its students to be active learners who participate in and are involved with the subjects being studied. This is especially true in the fields of science, where young students are encouraged to engage Biology, Chemistry, and Physics in very hands-on and interactive ways. Much more so than other subjects such as mathematics or history, science being taught to middle and high schoolers tries to emphasize interactive learning that surpasses one-dimensional memorization or regurgitation. The results are shown in the students tending to remember and enjoy the laboratory adventures of dissecting frogs, making soap, or measuring the friction of rolling objects more than any science lecture or homework assignment.

However, the laboratory experiments taught today fall short of their intention of interactive learning. Most of the experiments are static, where the students follow a set of procedures and observe the results that have already been seen and are dictated by the correct procedure. Although a notch above reading and memorizing the results of an experiment written in a textbook, it still lacks the sense of dynamic learning that is associated with true science. Current experiences are adventurous only because the student is hands-on with unfamiliar things (e.g. a preserved frog, Bunsen burner, or caliper), whereas true science is adventurous because it explores unknowns with the possibility of new discovery. Current experiments also usually aren’t engaging in the practical sense because they force every student to follow the same strict steps in the same order to learn the same principle, whereas true science allows and even encourages the freedom of decision-making and path-choosing as part of the learning process.
The Internet has grown exponentially since its inception, and has just started being discovered as an incredible resource for learning. It has boundless interactive audio-visual capabilities inherent to its technology, begging to be reined in and focused towards dynamic multimedia-based education. Its availability to the average American has risen dramatically and has become almost standard in its association with personal computers. In fact, new technologies such as WebTV, Internet appliances, Microsoft’s X-Box computer, and even game consoles such as Sega Dreamcast and Sony Playstation 2 are making American households more and more familiar with the Internet. Internet access has even started to become common in the American public school, something unheard of only a few years ago. This availability plus familiarity of the Internet with students in schools opens opportunity upon opportunity for technology to enhance or even replace current education methods.

BigSignal is a project that seeks to capitalize on this largely untapped potential of using Internet technology to help young students learn. This project applies the strengths of Web multimedia to provide students with the opportunity to do true science. It accomplishes this by linking students to a real science mission and allowing them to become the scientists on the mission. Specifically, BigSignal is tied to the NASA/CMU (Carnegie Mellon University) project RAMS (Robotic Antarctic Meteorite Search), which uses robot rovers to find and analyze meteorites on the Antarctic surface. In particular, BigSignal 2000 followed the Nomad rover, which has gone several times to Antarctica to search for meteorites, on its trip during January of 2000. The project brought the mission to several Pennsylvania middle and high schools through the Internet and involved the students as actual scientists on this actual mission.
Road To 2000 Project Deployment

Nomad
Scientists are trying to teach Nomad to autonomously do geology, from path-finding to marking targets to target acquisition to analysis of target to comparison of target with previous knowledge. Nomad is programmed to sweep a particular area, looking for isolated small dark objects against a white background. It selects one object, approaches it, takes field data (pictures, spectral readings, metal detector readings), and makes a conclusion about what type of rock it found based on comparisons with library data and previous field data, taking special note of those thought to be meteorites. The robot is attempting to do true science, making consequential decisions with the possibility of new discovery by remotely and autonomously doing what actual human geologists would do. BigSignal is matched with Nomad's deployment to uniquely provide a gateway through which students can follow the robot, see the same data, and make their own conclusions about the rocks targeted. Just as the robot is doing true science, so the students have the opportunity to do true science in the form of remote geology through the robot's perspective via the BigSignal Web interface.

Original BigSignal Project
BigSignal 2000 is the second iteration of the original BigSignal pilot project, done with the 1998 deployment of Nomad in Antarctica and three Allegheny County (Pennsylvania) high schools. The objective of this project was to allow students and teachers to participate in a telerobotics mission through Web-based education modules. It followed Nomad from November 1, 1998 to December 4, 1998 and gave students the opportunity to see the mission progress. The Web interface gave students access to an “I, Robot” ticker which personalized the robot’s accomplishments and decisions, a map showing the robot’s position on a transposed map of Antarctica, hires pictures of rock targets, and panoramic pictures of the view outside the robot.
Data was sporadically satellite-FTP'ed (File Transfer Protocol) to a computer server at CMU, then manually processed to Web-accessible format. Much of the data displayed in various plug-ins available as add-ons for current browsers to enhance the visual experience (e.g. QuickTime, VRML). Lessons learned from the shortcomings of the pilot project were to be applied to future deployments (i.e. BigSignal 2000). These were:

**Focus on middle schools instead of high schools**, because high school curriculum was too specialized to incorporate a multidisciplinary topic such as remote geology.

**Supplement classroom teacher curriculum with CMU teaching innovation**, because new teaching methods were needed to make the online tools into effective teaching tools.

**Create “classroom-proof” technology**, because the plug-ins were designed for current computer technology, not the computers present in the public school system.

**Send a BigSignal team member to Antarctica**, because the receiving of data was not timely and not supervised by someone with specific interests in BigSignal.

**Introduction To The 2000 Project**

The 2000 project officially started on April 1, 1999 at the Field Robotics Center on the Carnegie Mellon University campus (http://www.bigsignal.net). The core team consisted of a director, a software engineer, an interface/Web designer, a curriculum developer, a technical writer, planetary geologist, and a database/robotics engineer. I joined the project on June 1, 1999 as the lead software engineer. The target was to have a refined, technologically sound product as well as a robust curriculum ready to be employed at schools parallel to the January, 2000 deployment of Nomad in Antarctica.
To overcome the shortcomings of the original project, BigSignal 2000 was geared from the beginning for use in Allegheny County middle schools. The project also partnered with a group of researchers at CMU who apply cognitive psychology to enhance learning. One of the core team was designated to travel with Nomad to Antarctica. Similar to the original project, there would be many Web pages to describe necessary background information for the students to get involved. They would reveal the purpose of the mission, the target geographical area, the team members, the scope of the mission, even the algorithms used by Nomad, etc. The most critical part of the BigSignal 2000 curriculum would be the live mission interface through the robot console, where students would actually be linked with the robot as it did science.

The most difficult decision involved the technology to bring the robot console to the somewhat outdated computers in the public schools. The decision was that instead of making a Web page with many different parts (as in the original project), many which required a separate plug-in to be downloaded and installed, the entire interface would be made as a Java applet. Java was chosen as the programming language because of its platform independence, meaning equal accessibility from Mac, PC, or UNIX. Also, Java was unique in its graphical capabilities, with toolkits specifically designed for interactive multimedia that could encompass all desired functionality and look. Lastly, Java applets could be run from any browser without any download necessary, with Java runtime environment built into every browser.

I was responsible for engineering all parts of the Java applet except the back-end database that the applet accessed and the initial rock comparison tool (discussed later). I was also responsible for integrating and refining all components, as well as making sure that the applet was as efficient as possible. I worked closely with the interface designer and took advice from the human-computer interaction researchers to develop the applet’s look. I consulted the curriculum developer to incorporate desired functionality into the applet. I gave input to the database programmer to mold the architecture through which raw data from Nomad in Antarctica would be easily accessed and displayed by the applet.
BigSignal 2000 Applet

The BigSignal 2000 applet was developed in Java version 1.1.6 on a Unix platform utilizing the Abstract Windows Toolkit. The applet is accessible through http://www.frc.ri.cmu.edu/projects/bigsignal/2000/console/console.html. It totals over 3,000 lines of code; its different parts were developed as separate, stand-alone components before being put together. The applet interface is shown in Figure 1.

Figure 1: Applet interface

The coding of this applet can be broken into six parts:

- Database
- Timeline/Daily Reports
- Map
- Panoramic Picture Viewer
- Rock Comparison Tool
- Integration
Database
The database is a Java servlet that is spawned whenever its host Web server gets a specific URL request. The servlet has two tasks: transform raw data into Java objects accessible by the applet interface and communicate with the applet to transfer those objects. The raw data is in the form of database archive files, created by Nomad's science autonomy database. These archive files would be manually satellite-FTP'ed from the base camp in Antarctica to the host Web server at CMU hopefully once a day. A cron job written on the server would automatically wake up the servlet every twelve hours to process new raw data that might have come in. This database is completely automated. As soon as data has been sent from Antarctica and the cron job triggered, the database would handle the new archive and concatenate the new Java objects to the old ones (that way objects would be stored in chronological order). It would then be ready to dynamically respond to any requests the applet makes.

The main Java object that the applet accesses from the servlet is the DatabaseRecord object. It holds all images, rock analysis data, time stamps, and GPS (Global Position Satellite) coordinates (detailed description found in Appendix A) for each event recorded by Nomad. When the applet is first booted up, it loads in all the DatabaseRecords in the order they were put into the servlet (chronological order). It sequentially creates arrays and vectors (dynamic arrays) to store all images, data, positions, and timestamps for chronological access. Since the applet also requires geographical access to data (e.g. determining which icons to draw when the user is scrolling the map), it also makes reordered arrays and vectors according to position. It does this by dividing up the square representing the entire area searched into a 3x3 grid, and keeping a vector of DatabaseRecords for each portion of the grid. This provides for more efficient searches to determine which icons to draw in the overhead map view (more detail later).
All of this is pre-loaded by the applet and stored in memory. This causes a longer waiting period before the applet can actually run, but also makes interacting with the applet very fast once it has finished loading. The reason for doing it this way is that young students in this culture are very impatient, especially with electronics, so emphasis was placed on having the applet respond to user input as quickly as possible. Also, the decision factored in teachers being able to load up the applet before students got to class.

The first and most natural way of visually displaying the data is chronologically. This is done through the Timeline.

**Timeline/Daily Reports**

The Timeline displays the chronological attribute of the *DatabaseRecord* object. The Timeline/Daily Reports component is shown with its different parts in Figure 2.

![Figure 2: Timeline/Daily Reports](image)

The first part of the Timeline is the dynamic menu. By clicking on the items in the menu, the user can toggle on or off the icons associated with that menu item type. The toggling of icons occurs both in the Timeline and in the Map. This is useful for viewing the icons only of a certain type, especially if icons on the map of a different type overlap.
The second part of the Timeline is the chronological display of acquired data. The display shows a certain number of days as chosen by the user (discussed below) and can be scrolled through using the scrollbar on the bottom. Each day is denoted with a vertical line (representing noon of that day) labeled with the day of the mission (starting with 1). The middle day in the display is highlighted in black. The initial view of the timeline shows the most recent day of the mission highlighted in a view with seven days. Icons for the different data types are placed according to the time stamp field of their DatabaseRecord object. Clicking on an icon sends the appropriate data from that DatabaseRecord to either the Rock Comparison Tool or the Panoramic Picture Viewer. It also draws or changes the colored selection box to that icon, depending on its type.

There is a zooming panel above the chronological display that controls the number of days shown in the display. The initial view has seven days, and can be zoomed in to three or one, or zoomed out to fifteen or thirty-one. The zooming recalculates and redraws the display, centered on the highlighted day.

The Daily Reports section displays the text email for the highlighted day. The emails are sent from the field every day of the mission with the weather conditions and a summary of the day's events. They are manually entered into files on the server that the applet accesses each time the highlighted day changes, unlike all other field data, which is dynamically accessed.

The chronological display of data in the Timeline/Daily Reports is complemented by a geographical display of the same data in the Map.
Map

Although the data is stored chronologically, it is valuable for scientists as well as students to see the logistics of where data was acquired. This can reveal a cluster of meteorite finds as well as reveal the path-finding of Nomad. This is accomplished through the Map, as shown with its various parts in Figure 3.

![Figure 3: Map](image)

The first part of the Map is the actual overhead view. The initial view is the smallest square that contains all the events. The user can use the scrollbars to scan this view. Each time the user changes the view, the applet recalculates and redraws it. However, to make the search more efficient, the applet takes advantage of the data already being divided geographically into a 3x3 grid. Depending on the boundaries of the view chosen by the user, the applet only needs to search certain parts of the grid, instead of searching through all the DatabaseRecords every time. This is shown in Figure 4.
Figure 4: Applet utilizing grid for overhead map

Absolute position comes from the GPS attributes of a DatabaseRecord object and is shown through an (x,y) coordinate pair representing the meters from the base camp position in a particular direction. These coordinate pairs mark the position of the upper-left and lower-right corner of the view. Mouse clicks in the view have different results depending on what option the user has selected (discussed below). At the bottom portion of the overhead view, the position of a selected icon of each type is displayed.

The selection panel in the top portion of the Map has a button that can be clicked to restore the default view in case the user has lost orientation. The panel also includes a set of radio buttons which determine what function mouse clicks in the overhead view will execute. If the user has chosen select, a mouse click on an icon will send the appropriate data from that DatabaseRecord to either the Rock Comparison Tool or the Panoramic Picture Viewer. The proper colored selection box will be drawn or changed to that icon. If + or – is chosen, a click will zoom in or out on the overhead view, recalculating and redrawing the view centered on the clicked location.

In a small, attached panel, there is a thumbnail map. This map helps orient the user by showing the overall, highest-level view with all icons as dots. Whatever area the current overhead view encompasses is demarcated by boundary lines on the thumbnail.
The Timeline and Map organize the data in such a way that orients the user to the available information. The Rock Comparison Tool and Panoramic Picture Viewer components actually expose the details of what that information holds.

**Rock Comparison Tool**

The Rock Comparison Tool is where the science actually takes place. This component is shown in Figure 5.

![Figure 5: Rock Comparison Tool](image)

The first part of the Rock Comparison Tool is the rock library on the left. These are many common rocks as well as previously found meteorites categorized by family (see detailed description in Appendix B). The interface has a list to select the family and a set of `ImageButton` s to choose which library rock to view. When a family is chosen or rock is clicked, the applet jumps to right place in the file structure and pre-loads all necessary pictures before drawing them on the screen. The library serves as a reference to help the user classify unknown field rocks, with comparison enabled through visual analysis (hires pictures) and rock attribute analysis (spectral and metal detector data).
The comparison is actually done in the two *ImageViewer* panels adjoining the rock library. On the left is the selected rock from the library and on the right is the field rock selected from the Timeline or Map. By clicking on the tabs on the top of the panels, the user can switch between comparing pictures, spectral waveforms, or metal detector waveforms. The picture of the library rock can be animated for full 360 degree viewing of the rock by scrolling through several frames (totaling seven) via the scrollbar at the bottom of the panel. The colors surrounding the pictures correspond with the colors of the waveforms, which correspond appropriately to either the rock library or Timeline/Map selection box color.

The remote science done in the Rock Comparison Tool would be significantly enhanced by a true sense of immersion in the surroundings. The sense of adventure and intimate involvement with the mission comes from the Panoramic Picture Viewer.

**Panoramic Picture Viewer**

The Panoramic Picture Viewer is shown below in Figure 6.

![Panoramic Picture Viewer](image.png)

**Figure 6: Panoramic Picture Viewer**
Nomad takes panoramic pictures using a pano-spherical camera. Because the camera does not take flat, two-dimensional pictures, a de-warping algorithm triggered by the cron job transforms the original to a flat picture. The user can scroll up to and beyond a 360 degree view around Nomad by repeatedly clicking either the left or right blue arrows on the side of the panel. The green words on the bottom of the panel give orientation to the user as to what angle the picture is showing relative to Nomad. The ability to continuously scroll in one direction is achieved by drawing two identical pictures adjacent to one another. When the user scrolls past one of the ends of the pictures, the view jumps to the edge of the other picture, giving the impression of continuity. This is demonstrated in Figure 7.

**Figure 7: Continuous scrolling for panoramic pictures**

![Applet view](scrolling_right)

Each component was developed and tested separately, then integrated.
Integration

The integrated applet interface was shown in Figure 1 above. Each component was developed as a separate frame, so integration involved making each individual frame into a panel of one big frame. The size of the big frame was decided to be slightly smaller than 800x600, since that was the normal viewing resolution of the monitors at the schools. The layout links the Timeline and Map together and the Panoramic Picture Viewer and Rock Comparison Tool together. Each component has a label, and all parts of components that are related are associated by color. This is true for background colors as well as surrounding selection box colors.
Problems And Solutions

By December of 1999, Nomad was dissembled and shipped to Antarctica. Testing and refining of the applet was underway, and curriculum was being finalized. Trips were being made to the schools as part of the applet testing process. Also, before Nomad started its actual mission, dummy data archives were sent back from Antarctica to test the servlet and its interaction with the applet. In that process, four major problems arose:

- **Dynamic data errors**
- **Memory consumption/applet speed**
- **Java 1.1 incompatibility**
- **Inconsistent browser handling**

**Dynamic data errors**
To test the database, a manually created archive was uploaded to the server, the cron job was triggered, and the applet accessed the Java objects from the servlet. However, what was not tested was the same process with dummy archives created by Nomad, as well as with archives added after this process had finished once. Nomad actually saved each archive chronologically, but it stored the data in the archives in reverse-chronological order. When the archives were FTP’ed back the server, they were correctly processed by the server and applet, but interfered with the applet’s display and calculations, since the applet had assumed chronological order. Also, after one round of processing archives, the servlet, when triggered by the cron job, would not properly process the new archives.
This problem was solved through hard-coding all numerical data (i.e. GPS positions, time stamps) into the applet. This new applet was created through stopping the servlet, deleting all unarchived data, reinserting all previous archives, starting the servlet, and manually triggering the cron job. The numerical data was then accessed by the original applet and manually copied to the new applet. This ensured chronologically ordered data, as well as proper processing by the servlet because each time it only had to process one set of archives.

**Memory consumption/applet speed**

When testing the applet on the school sites, it consumed too much RAM (computer memory) and ran too slowly (not unrelated problems). These problems were mainly caused because of the pictures Nomad was storing. Unlike standard Web pictures stored as small .jpg or .gif files (~10 Kilobytes each), the pictures from Nomad were stored as .ppm files, simpler but tremendously larger in size (>1 Megabyte each). When the applet went to access these pictures, speed and memory became deficient.

To solve this problem, each of the .ppm files was manually converted and compressed into .jpg files. They were placed in a directory on the server and hard-coded into the applet as part of the pre-loading process. That way, loading time for the applet was increased, but speed and memory were significantly less deficient (a desired trade-off, as discussed on page 8).

**Java 1.1 incompatibility**

The applet was developed in Java 1.1, but the older browsers on many school computers could not run Java 1.1 properly. Because the target of the project was easy access for the public, the solution was not to download new browsers. Instead, the applet needed to be recast into Java 1.0 (the original version of Java that older browsers can handle).

This was accomplished through downloading a Java 1.0 compiler and compiling the Java 1.1 applet’s code. The resulting list of errors was worked through one by one, rewriting the functionality of Java 1.1 commands from the applet into Java 1.0 terminology.
Inconsistent browser handling

The final major problem encountered resulted from using the applet on different browsers and different platforms. Although Java is supposed to be platform independent, testing of the applet exposed this as a weakness of Java. Netscape Navigator and Internet Explorer did not display the applet the same way; significant differences were detected between the Macintosh and PC (and other non-Macintosh platform) tests as well. Because the differences between the applet run on the Macintosh and run on other platforms were mainly cosmetic, two different versions were made so that the user's experience on any platform was the same. The differences between Netscape Navigator and Internet Explorer could not be reconciled, since some of the older Navigator browsers just have inferior implementation of Java. Some examples of quirky behavior in Netscape include improper handling of scrollbars, unnecessary and uncalled for applet repainting (causing flickering and pauses), and occasional glitches in displaying pictures.
The BigSignal 2000 project was not intended as an end in itself. Although a second iteration of the original BigSignal project, this was still a pilot project, made obvious by its short timeframe. It was a stepping stone project for the true goal of involving students from around the globe with real scientists to make real discoveries in extraterrestrial planetary exploration.

**2000 Project Deployment Summary**

The mission lasted nineteen days, starting January 12 to January 30 of 2000. There was a total of twenty-five hires pictures and eighteen panoramic pictures sent back from Nomad during that time. Three of the rocks found were definitely meteorites, several others were thought to be meteorites and sent to NASA for further testing. Nomad become the first robot ever to discover a meteorite completely autonomously.

Because of the pilot nature of the project, there were changes to the applet in the duration of the mission. Because Nomad did not take along its metal detector, the ability to compare metal detector data was removed. Also, the spectral data taken by hand for the library rocks was not comparable to the spectral data taken by Nomad because the placement and proximity of the spectral analyzer relative to the rocks was different. This functionality was also removed. For the sake of increasing the speed of the applet, the daily reports were reduced from full pages-long text to only the lines of the email that described the weather conditions.

The applet and week-long curriculum were deployed in four classrooms: seventh and eighth grade of Peters Township Middle School, seventh grade of Greenfield Middle School, and six grade of Quaker Valley Middle School (curriculum in Appendix C, pictures in Appendix D).
Future Projects

The next step for BigSignal 2000 is to create a commercialized classroom module. This module would be completely stand-alone and distributed in hard copy to classrooms nationwide. Even though the live mission is over, there is still much value to the learning experience of the 2000 project. Even after data gathering has been completed, there is still science to be done. Data processing and gleaning after the mission is done is an important aspect of remote science.

The project which both BigSignal projects were pilots for is EventScope (http://www.eventscope.org). This is another joint NASA/CMU project already approved and underway, with the purpose creating a completely new tool for students to visit, explore, and contribute to science being done by rovers on other planets. As quoted from the Web site: “The visual interface will simulate a telerobotics control room, with each student or student team operating a simulated robot modeled on the one NASA used in that module’s mission. Each student or team will interact with an onscreen team of characters representing professional scientists and engineers. These characters will convey contextual information related to the mission, as well as linking to information about their specialties. Students will be able to ask these characters questions via email, receiving answers that are actually from participating professional scientists. At specific milestones, students and teachers will be able to share information with other classes involved in the same modules. A decentralized community of students, educators and scientists will be created.”

EventScope delves into new dimensions of involving students in actual science missions. Even beyond the limited scope of science done in parallel with Nomad, the science done by students in EventScope will have unlimited scope, depth, and breadth. Students making discoveries will be a common occurrence, especially with the help of real NASA scientists who are looking at similar data. Because the data set gathered will be so large, middle and high school students will actual contribute to NASA’s mission by providing that many more pairs of searching and hungry eyes.
Conclusions

The Internet is an emerging technology that has incredible potential to help students learn due to its audio-visual and interactive abilities. As it becomes more and more accessible and as its bandwidth to schools increases, it becomes more and more of a natural match with science education, both in an enhancing and replacing role.

BigSignal 2000 was an ambitious pilot project pointed towards a goal of using the Internet and its tremendous resources to immerse middle and high school students in true science missions and projects. The project was progress towards this end because it linked Pennsylvania middle and high schoolers to the Robotic Antarctic Meteorite Search, an actual NASA/CMU science mission. It gave students the opportunity to look at all kinds of data being taken by Nomad in Antarctica and make their own conclusions. There was no direct contact with NASA scientists and no decision-making by the students that would affect Nomad. These would be additional features to be implemented in EventScope, the project that the BigSignal projects were piloting towards.

For the scope that BigSignal 2000 was targeted, the project was a major success. Main areas improved from the original BigSignal were more complete and innovative curriculum, better technology making for easier classroom deployment, and excellent contact with and greater immersion in the mission in Antarctica. Each of these were goals stated at the beginning of the project (discussed on page 4), making this pilot project a big stepping stone for EventScope and a great success.
Works Cited


Appendix A: DatabaseRecord Object

public class DatabaseRecord implements Serializable, Cloneable {
    // These constants are used to figure out the type of sensor data.
    public static final String HI_RES_SENSOR_SAS_NAME = "smHiResDriver";
    public static final String ARM_SENSOR_SAS_NAME = "smManipulatorDriver";
    public static final int HI_RES_IMAGE_NUM_BYTES = 1000000; // ~1MB
    public DatabaseRecord cloneRecord();
    public boolean addSensorData(String sensorName, String filename,
                                  TimeStamp t) throws FileNotFoundException, IOException;
    public String toString();
    public int getTargetID();
    public Image getHiResImage();
    public Image getArmCamImage();
    public Image getPanoCamImage();
    public Spectrum getSpecData();
    public Classification getClassification();
    public float getClassProbability(String name);
    public float getDGPS_X();
    public float getDGPS_Y();
    public float getDGPS_Z();
    public TimeStamp getAcquisitionTime();
    public TimeStamp getHiResTime();
    public TimeStamp getArmTime();
}

http://www.bigsignal.net/BigSigDatabaseServlet.html
## Appendix B: Rock Library

<table>
<thead>
<tr>
<th>Family</th>
<th>Rock Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Igneous</td>
<td>Ash flow tuff</td>
</tr>
<tr>
<td></td>
<td>Augite</td>
</tr>
<tr>
<td></td>
<td>Basaltic glass</td>
</tr>
<tr>
<td>Metamorphic</td>
<td>Meteorite impact material</td>
</tr>
<tr>
<td></td>
<td>Shatter cone</td>
</tr>
<tr>
<td></td>
<td>Schist</td>
</tr>
<tr>
<td>Mineral</td>
<td>Realgar</td>
</tr>
<tr>
<td></td>
<td>Covellite</td>
</tr>
<tr>
<td></td>
<td>Serpentine</td>
</tr>
<tr>
<td>Sedimentary</td>
<td>Sandstone</td>
</tr>
<tr>
<td></td>
<td>Mudstone</td>
</tr>
<tr>
<td></td>
<td>Dolomite</td>
</tr>
<tr>
<td>Volcanic</td>
<td>Andesite</td>
</tr>
<tr>
<td>Meteorites1</td>
<td>Kulnine 1</td>
</tr>
<tr>
<td></td>
<td>Kulnine 2</td>
</tr>
<tr>
<td></td>
<td>Carraweea 1</td>
</tr>
<tr>
<td></td>
<td>Carraweea 2</td>
</tr>
<tr>
<td></td>
<td>90524</td>
</tr>
<tr>
<td>Meteorites2</td>
<td>ALHA 81008</td>
</tr>
<tr>
<td></td>
<td>Huckitta</td>
</tr>
<tr>
<td></td>
<td>Sikhote 1</td>
</tr>
<tr>
<td></td>
<td>Sikhote 2</td>
</tr>
</tbody>
</table>
Appendix C: Classroom Curriculum

Schedule for Classroom Activities:

Prior to Day One: General mission description, reading. Present basic information on robots, Antarctica, and meteorites. Distribute handout covering this information to students to read as homework. No writing. Just some free thinking, discussion and exploration of the topic.

Day One: Brief Introduction (5). Creation of teams (role description, self-nomination, team assignments) (5 to 10). Re-arrange classroom into labs, test field for the robot, and team areas (5; this could also be done prior to class). Distribute equipment and reading materials (2). Robot Control Run 1 (total 20 minutes, 3 min. each team maximum). Give teams some free time to talk about their plans for the week (rest of period).

Day Two: Robot Run 2 (15 minutes, more organized than day 1). Introduction to the Web Site and Tools (10). Free-team time. Start the Search. (Rest of period.)

Day Three: Robot Run 3 (20 minutes, harder run). Computer time for meteorite search, and/or writing/team discussion, as resources allow (rest of period).

Day Four: Robot Run 4 (20 minutes). Computer time for meteorite = search, and/or writing (rest of period).


After 5-Day: Each team must produce a team report, including the robot report, the geological report, the computer report, the project report, and organized notes/minutes from the reporter.

http://www.frc.ri.cmu.edu/projects/bigsignal/2000/classroom/classroom.html
Appendix D: Pictures From Deployment

Peters Township Middle School (Eighth Grade)
Big Signal's first pilot school to complete the program!