Skyward

Requirements Specification Report

12/7/2016

Version: 1.1

Team Name:

Skyward

Team Members:

Gage Cottrell
Kaina Crow
Justin Kincaid
Chris French
Alexander Sears

Sponsors:

Dr. Michael Mommert and Dr. David Trilling

Mentor:

Dr. Otte

Acceptance Statement:

Accepted as baseline requirements for the project:

____________________ (Date) _______________________ (Client Signature)

____________________ (Date) _______________________ (Team Lead Signature)

Table of Contents

Introduction

Near Earth Objects (NEO)

Everyday about one hundred tons of particles originating from outside of Earth's atmosphere makes contact with Earth's surface. While the vast majority of the particles contacting Earth are very small, instances of larger objects impacting the surface have been recorded. Roughly every 10,000 years, a rocky or iron asteroid larger than 100 meters impacts the earth and causes localized destruction. On an average of every few 100,000 years, asteroids larger than 1 kilometer impact the earth and cause global disasters that can severely impact life on Earth. Due to the risk of having Earth impacted by objects that have the potential to cause massive destruction, it is vital that near earth objects be discovered, tracked, and analyzed.

Researching Near Earth Objects (NEOs) is an ongoing joint effort between many countries and governments. Using telescopes, it is possible to discover and track NEOs as they move throughout the solar system. Making publically available the information gained from NEO discoveries, allows for a collaborative environment in which NEOs can be discovered, tracked, and analyzed very quickly. To date, there have been more than 15,000 different NEOs discovered. As telescope technology advances, so do the amount of NEOs discovered. Figure 1 below depicts the discovery rate of near-earth-asteroids from 1980 to present date.

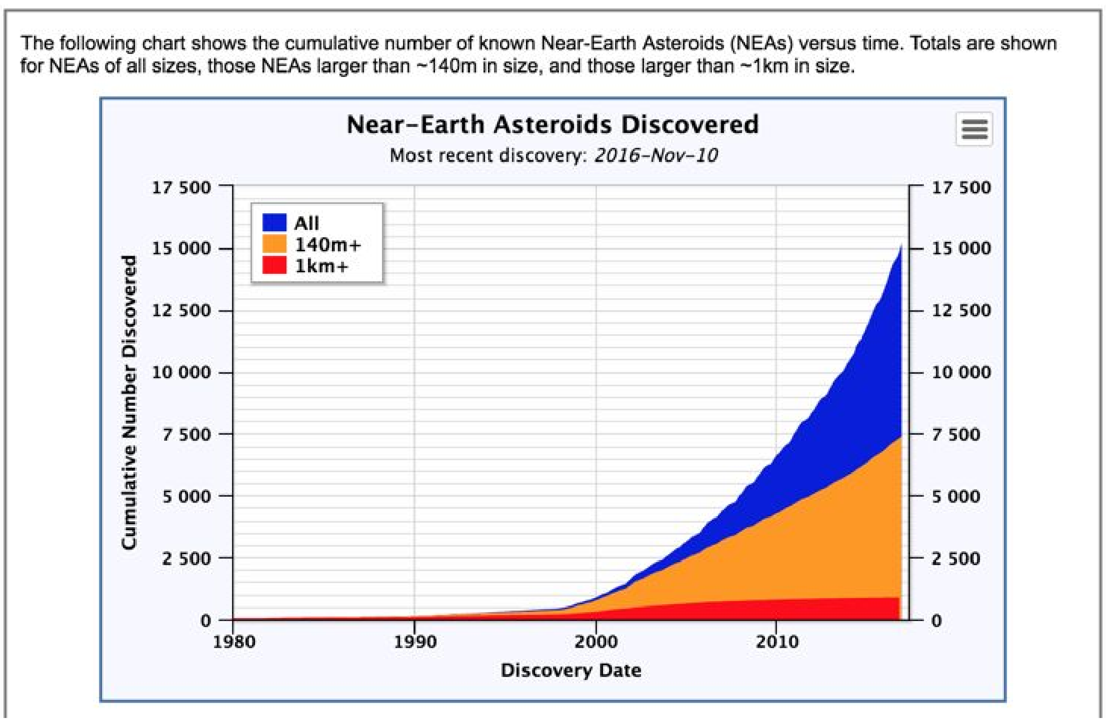


Figure 1: Near-Earth Asteroids Discovered (Source: http://neo.jpl.nasa.gov/stats/)

The North American Space Agency (NASA) has been actively working towards the discovery of additional NEOs. The 'NASA Authorization Act of 2005' allowed the agency to "implement a program to detect, track, and catalogue the physical characteristics of Near-Earth objects equal to or greater than 140 meters in diameter". The act also established a goal of discovering over 90% of these NEOs by the year 2020. When the authorization act was established in 2005, the budget for the resulting NEO project was $4 million. This budget increased to over $40 million by 2014. NASA has already stated that it will not be able to meet the 90% catalogue goal by 2020.

With funding from NASA and from the National Science Foundation, the Department of Physics and Astronomy at Northern Arizona University (NAU) has begun work on repurposing a telescope to help aid with the analysis of newly discovered NEOs. The Flagstaff Robotic Survey Telescope (FRoST) is the telescope in question, and NAU has already received more than $3 million in funding from NASA to support the project. Once completed the telescope will be able to remotely capture images of newly discovered NEOs and share its data for analysis with outside sources. The Flagstaff Robotic Survey Telescope aims to visualize all candidate NEOs brighter than the 20th visual magnitude, within one hour of the object's discovery.

The leaders of the FRoST project are Doctors Michael Mommert and David Trilling. Dr. Michael Mommert received his Ph.D. in geophysical sciences at Freie Universität Berlin, Germany, in 2013. Dr. Mommert's areas of focus are primarily Near-Earth objects and Outer Solar System bodies. Dr. David Trilling is an Associate Professor at Northern Arizona University, and received his Ph.D. in planetary science from Arizona State University in 1999. Dr. Trilling's areas of study are: the evolution of our Solar System, observational planetary astronomy, studies of other planetary systems, and data mining.

Skyward Project Clients

Doctors Mommert and Trilling have worked on many projects together previously such as: observing NEOs with the Spitzer Infrared Telescope, focused on rapid-response color observations of NEOs with ground-based telescopes, and upgrading a thermal-infrared camera on a telescope in Hawaii. Although many of Doctors Mommert and Trillings projects have been on teams comprising of about ten people, their research benefits 7.4 billion people since they are helping to characterize potential impactors and the overall near-Earth asteroid population.

While the FRoST team has all the qualifications and experience required in order to complete the goals of the FRoST project, there are a few remaining problems and challenges that remain to be solved before the telescope becomes operational. The following section will describe a few of the challenges that remain to be solved before the launch of the FRoST project.

Problem Statement

Current problems for telescope users

Almost every modern telescope has a way for users to view important information that includes weather conditions around telescope, current position of the telescope, and current all sky view. Having this data easily accessible allows for users to check on the status of the telescope and ensure that it is operating properly. As of this moment, the FRoST project has no way of allowing users to view such information.

Currently, the FRoST project team must physically visit the telescope in order to view any information about its current status and surrounding weather conditions. Not having a user constantly on location means that someone must take valuable time out of their day to travel to the telescope to view the status. Due to the risk of rapidly changing weather conditions or technical malfunctions that can arise when working with a telescope, not having the latest telescope and weather data is a major problem and possible liability.

Current problems with telescope communication

The telescope receives commands from Lowell observatory in Flagstaff, Arizona via a secure network connection. Once commands are received from Lowell, the telescope sends acknowledgement statements back to Lowell, so that users can confirm that the telescope did in fact receive the commands successfully. After sending acknowledgement statements to Lowell, the telescope executes the commands it was issued. However, just because Lowell receives acknowledgments from FRoST does not mean that the telescope executed its issued commands successfully. This could lead to a major issue where users of the telescope believe that the telescope executed its issued commands successfully when it did not. For example, if Lowell issues a command to FRoST telling it to look at a specific position of the night sky, and the dome surrounding the telescope fails to open, users would believe that everything is operating as normal when in fact there has been a failure. If this was to happen, the users would only know of the failure once they realized the images that the telescope is taking are of the inside of the dome! Only then would users realize the failure and be dispatched to fix the problem on site. This problem is a major challenge that must be solved before the telescope becomes operational.

Solution Vision

Our solution is an interactive web application

Team Skyward will work directly with the FRoST project in order to design and implement a solution to the discussed problems above. The solution that team Skyward has come up with is to create a dynamic web application that allows users to visit a web page and view the current status of the telescope, surrounding weather data, and other important information online. This web application will be known as the "FRoST Monitor". Creating FRoST Monitor to display the vital information that users want and need to see regarding the telescope saves the users of the FRoST project from having to physically visit FRoST to check on its status. The proposed web application will allow for an easy way to view the telescopes status and allow for a way for users to confirm that the telescope is operating as it should.

- ***Specifically, the FRoST Monitor application will feature the following capabilities:***
    - Receive and display current telescope status information
    - Receive and display current surrounding weather data
    - Receive and display up to date images from Lowell Observatories "All Sky Camera"
    - Allow authenticated administrators to remotely shut down the telescope in case of emergencies

FRoST web application will save time and money

The FRoST Monitor project has the potential to save hundreds of hours of travel time to and from the telescope, as well as providing a way to quickly warn the FRoST team of potential damage to the telescope in the event of a system failure or bad weather - saving thousands of dollars. Once the FRoST Monitor software is completed and implemented, the FRoST project will be one step closer to being completed and operational. Team Skyward is very confident that we will be successful in designing and implementing the FRoST Monitor system. The key functional, performance, and environmental requirements of the FRoST Monitor system will be discussed in detail in the following section.

Project Requirements

These are the requirements gathered from meetings and discussions about the Frost Monitor project. The requirements are presented as interpreted by Team Skyward. Discussions between team members were used to determine the best way to annotate the desired outcome of this project.

User domain level requirements

- ***UD-1.*** *Access Lowell weather station information:*
    - Gather information on current weather conditions of the Mesa Observatory.
- ***UD-2.*** *Website to display telescope status and weather:*

- Web application to provide access to telescope status data

- ***UD-3.*** *Single password protected user:*
  - Provide a secure login tunnel for elevated privilege access to web application.

- ***UD-4.*** *Broker communication between 'Nuthatch' database server and web application:*
  - Server side applications to send data from 'Nuthatch' machine to backend web application

- ***UD-5.*** *Mirror 'Nuthatch' Database to web server:*
  - Store website-relevant information on web server database.

- ***UD-6.*** *Curate telescope status information:*
  - Keep track of telescope vitals such as telescope pointing, timestamps of telescope information, target's angular distance to the moon, dome status.

- ***UD-7.*** *Maintain list of last issued commands to telescope:*
  - List of recent commands sent to telescope in a Log file.

- ***UD-8.*** *Shutdown button to send signal to Mesa observatory:*
  - Password protected web application interface to send a shutdown command to the FRoST Telescope.

- ***UD-9.*** *Access all-sky camera on Mesa Observatory:*
  - Retrieve images from all-sky camera and display on website.

Functional Requirements

- ***FR-1.*** *Web application:*
  - Web application will display up to date telescope status, local weather information, and NEO target information on the front page (as defined in user domain level requirements).
  - The Web application will be implemented using Django framework.
  - The front end web application interface will be implemented using responsive web design practices to be compatible with mobile devices.
  - There will be uniform style and appearance throughout the web application interface.
  - The web application back end will maintain a database for telescope status, local weather information, and NEO target information .

- ***FR-2.*** *Single Password protected user:*
  - The web application will provide a login page for an elevated privilege user.

- The link to the login page will not be displayed on the front page.
- There will be one administrative account provided.
- The account will use a username and password authentication using Django built in authentication and authorization system.

- ***FR-3.*** *Broker communication between 'Nuthatch' database server and web application:*
  - Server side applications will be implemented to communicate between 'Nuthatch' and the frost monitor application.
  - A REST API will be provided by the Frost Monitor application that will allow for pushing and pulling formatted data models from the database maintained by the web application.
  - A server side application using Python 3.* will be implemented on 'Nuthatch' to update web application database using the REST API provided by the web application using json formatted objects.
- ***FR-4.*** *Database to store mirror of 'Nuthatch' observational data:*
  - A sqlite3 relational database system will be used for the database.
  - The backend application will be implemented in Python 3.* and Django framework.
- ***FR-5.*** *Curate telescope status information:*
  - A server side application using Python 3.* will be implemented on 'Nuthatch' to collect telescope statuses, local weather information, and NEO target information from the Lowell network.
  - Data will be gathered from Lowell observatory from the server side application on the 'Nuthatch' machine.
  - The data will be stored on the 'Nuthatch' machine in a relational database.
  - The telescope metrics that will be tracked and logged are the telescope position, timestamp of received information, angular distance to the moon, the dome status.
  - The information will be relayed from 'Nuthatch' to the Frost Monitor web application through the REST api.
  - The telescope metrics will be displayed in real time on the front page of the web application.
- ***FR-6.*** *Maintain list of last issued commands to telescope:*

- Issued commands to the Frost telescope will be logged in 'Nuthatch' database.
- 'Nuthatch' will gather the last issued commands from Lowell network using a server side application written in Python 3.*.
- The commands will be sent to the web application using the REST Api.
- Commands from the last 24 hours will be displayed on the Frost Monitor application web interface.

- ***FR-7.*** *Shutdown button to send signal to Mesa observatory:*
  - An interface to relay a shutdown signal to the Mesa observatory will be implemented on the Frost Monitor application.
  - The interface will only be available to an authenticated user that is currently logged into the web application using the secure login page.
  - The secure shutdown application will only provide a way to send a shutdown packet through a secure connection.
  - The application will create a ssl tunnel and send a formatted tcp packet.
- ***FR-8.*** *Access Lowell weather station information:*
  - The current weather is provided from a weather station provided by Lowell observatory on the Mesa.
  - Current weather information will be collected from the Lowell network using a server side application on the 'Nuthatch' server written in Python 3.*.
  - The weather information will be parsed on 'Nuthatch' and posted to the Frost Monitor application REST api in a json formatted object.
  - The weather information will be displayed on the front page of the Frost Monitor website front page.
  - A 24 hour weather history will be provided along with current conditions and 48 hour and 3 day forecast provided by wunderground.
- ***FR-9.*** *Access all-sky camera on Mesa Observatory:*
  - Current all-sky camera images are provided from Lowell observatory from the Mesa.
  - Current all-sky images will be collected from the Lowell network using a server side application on the 'Nuthatch' server written in Python 3.*.
  - The images will be stored on 'Nuthatch' and their URLs with timestamps will be logged in the 'Nuthatch' database.

- The all-sky information will be parsed on 'Nuthatch' and posted to the Frost Monitor application REST api in a json formatted object.
- Current all-sky image will be displayed on the front page of the Frost Monitor website front page.

Performance Requirements

- ***PR-1.*** *Website must be available in the public domain.*
- ***PR-2.*** *The user must be able to login within 60 seconds upon landing on the login page.*
- ***PR-3.*** *'Nuthatch' should be able to communicate to the Frost Monitor application no matter what network it is connected.*
- ***PR-3.*** *REST API will enable all parts of the gathered data to be accessed and updated.*
- ***PR-4.*** *Information on telescope will be gathered from Lowell once every minute and immediately updated on front page.*
- ***PR-5.*** *Target list will be displayed on the front page; list of commands from last 24 hours will be accessible from front page.*
- ***PR-6.*** *Command list will be updated once every minute.*
- ***PR-7.*** *Current weather information will be gathered every 60 seconds.*
- ***PR-8.*** *Current all-sky image will be gathered every 60 seconds or according to Lowell all-sky image schedule.*

Environmental Requirements

- ***ER-1.*** *Deadline for the project is April 2017:*
  - The project requirements must be completed before the end of spring semester of 2017
- ***ER-2.*** *Client Server Nuthatch will store all required web application weather data:*
  - All of the weather data for the web application will come from the telescope on the Mesa and stored on Nuthatch.
  - Web Server data can only be as up to date as the information stored on Nuthatch
- ***ER-3.*** *Web server will be developed in a Linux server environment:*

- The Frost Monitor application must be deployed on the Northern Arizona University physics department web server.
- The web server is running a linux environment.
- The web server will be updated to the latest version of Red Hat O.S in December 2016.
- Django framework must be deployed on the physics webserver.
- Configurations files must be correctly implemented to allow for a python application to be served from Apache.

- ***ER-4.*** *Database will be a relational database implemented in SQL Lite 3.*

Potential Risks

Website could report inaccurate information

Our goal here is to ensure that the website will report the accurate state of the observatory. If the observatory dome is physically open, then this should be represented on the website. Other pieces of information includes the telescope orientation as well as the position of the dome itself.

Inaccurate or misleading information on the website may actually have more damaging results than first realized. From the research we have done so far, the general rule of thumb seems to be "if in doubt, close the dome". The physical telescope has many parts that could be potentially damaged if the dome is left open in case of precipitation. Even if the telescope is undamaged, humidity or precipitation could leave residue on parts of the telescope, resulting in blurry or otherwise inaccurate images. It is very important that the website accurately report whether or not the dome is actually open.

The website should also report the orientation of both the telescope and the dome. If the telescope and dome are misaligned, then the telescope will return black images, because it is shooting the ceiling of the dome.

Breach in security could result in damaging commands given to telescope

For the time being, it seems that the design will allow certain pieces of information to be viewable by the general public. The website will also allow certain commands to be given to the telescope, and it would be best if these commands are only given to privileged users.

One such command is the ability to shut the observatory dome. The option to shut the dome remotely would allow would offer an extra layer of protection from the telescope. If the dome were to close unexpectedly it could result in an entire night of wasted data. Any extra fuss over a potential misuse of the emergency dome closing feature is a simple fix by an added layer of security.

Another command that should be left away from the public is the ability to manually update the queue the telescope will use to decide what will be observed next. It is unknown at the moment whether or not the queue will be available to the public, but manipulation of the queue should be left out of their hands. It may not be too damaging if certain objects in the queue were moved around for the evening, but you would not want the entire queue to potentially be deleted.

A layer of security for these features is obvious, but it must be done in a way that mitigates potentially unwanted access to certain parts of the website.

Telescope Inoperational

A fairly large risk to the project is that the telescope could be inoperational by the time we finish this project. The graduate student that is in charge of implementing the software for the telescope itself could end up creating faulty software, which will affect how our project operates. There are also random risks such as malfunctioning telescope parts, or damage due to water or fire. All of these could severely affect how our project will be used in the future.

Project Plan

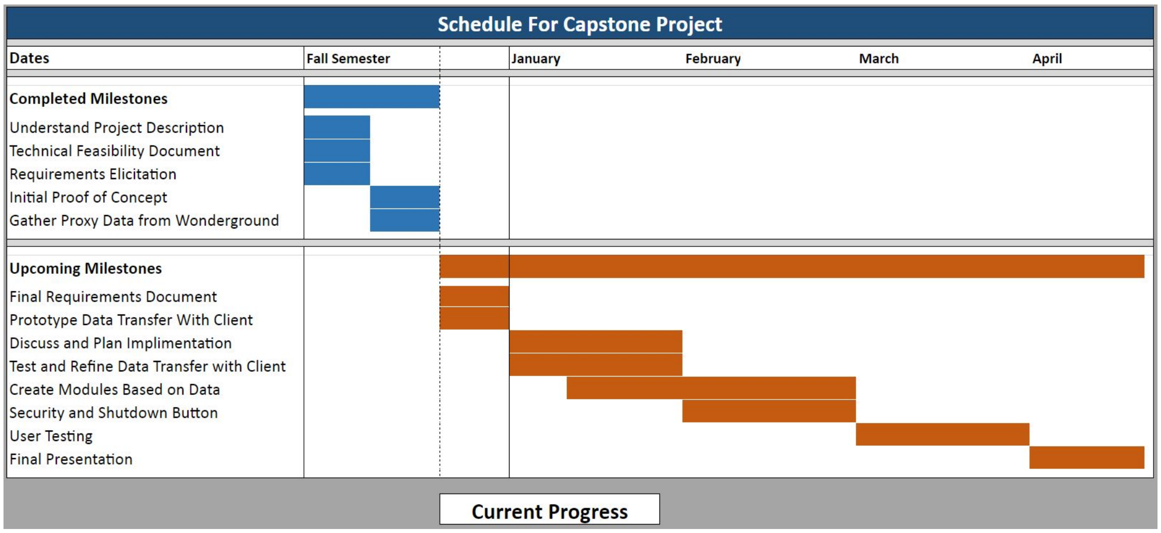
Nuthatch server

To start, there are basic pieces of information that will be captured from the telescope and Lowell observatory that must be displayed on the website. Before the end of the first semester, small proofs of concept must be developed to show that the communication needed for this data transfer is possible. Once this is done, the rest of the project involves handling gathered data and optimization.

Weather information and images will be captured by various instrumentation located at the FRoST monitor site on Lake Mary road in Flagstaff. This information will be transmitted to the Lowell observatory site North of Flagstaff, which will then be transferred to a server located on NAU campus. This NAU campus server is being developed by our clients, and is not quite finished at the moment. This server's name is Nuthatch, and it will be where we will be gathering all of the information for our web application.

Basics of project plan

The following chart showcases the basics of the project as it stands right now. The "Completed Milestones" shows a few items that have already been finished, including a small proof of concept consisting of making a simple web interface. The "Upcoming Milestones" section shows a few more items that must be completed before everyone leaves for winter break, as well as a breakdown of the larger tasks that must be completed before the end of next semester, including a proof of concept consists of gathering proxy weather data from another website until Nuthatch is finished. In the chart, these are larger grain problems, but each item in the list could be expanded to show several smaller steps. One example is "Prototype Data Transfer With Client", which is actually a summarization of implementing both sending and receiving data, which are actually two completely separate problems.



Proof of Concepts

Proof of concepts are pretty straight forward. Telescope information will be stored on our clients server, Nuthatch, and must be gathered to be displayed on the web application that we will be providing.

Nuthatch is not complete just yet which means that our proof of concept is gathering dummy data from a different website for testing. After Nuthatch is complete, we will adjust our web application to begin gathering data from the proper location.

Implementation

At this point, the communication between our web application and the Nuthatch server should be complete, which means we can begin working on several different items to complete our web app.

Improve weather data gathering:
Optimally, the information presented on our web app should be updated as quickly as possible. We have no control over how data is gathered from the telescope and uploaded to Nuthatch, which means our web app can only be as up to date as the Nuthatch server will allow. Because of this, any improvements we make to the speed in which data is displayed on our web app may not be immediately recognized. However, this will allow for any improvements to the telescope/Nuthatch communication to appear on our web application. This is not to say that there is any inherent problem with the telescope/Nuthatch communication, but if further

improvements are made outside our scope of the problem, we would like the web app to be adaptable.

Weather viewing modules:
Our proof of concepts will already be pulling data from Nuthatch, but we will have to find ways to make the data queryable, as well as display the data in a meaningful way, such as in the form of graphs or charts showcasing things like predictive weather, or just simple modules for current weather or weather history.

Features behind a security wall:
A single user account will be given to the client that will allow them to access several features that include and emergency shutdown button, and the ability to update the telescopes observe queue. Both features must be able to push data back to Nuthatch. This could be in the form of writing or updating a json file, but a clear solution is not present at the moment. We will first need to find a way to manipulate files on our clients Nuthatch server, then communicate with the client to establish a format for the data being transferred.

Updating the queue for the telescope may involve manipulating a queue file that already exists and is being used to submit commands to the telescope, but the command to close the dome may be a different solution altogether. Basically, both of these features are planned, but have not been researched quite yet.

Conclusion

Overview

The mission of the FRoST monitor project is an important one. The work that is being done to repurpose the Flagstaff telescope and to improve the manner in which weather information from the telescope may be monitored will lend to the effort to catalogue even more Near Earth objects of interest. As stated above, NASA has already realized that they will not reach their goal of cataloguing 90% of all NEOs greater than 140 meters in diameter by 2020, but perhaps the work that is done here may help reach the goal of 90% sooner than expected.

As the telescope stands right now, any weather information gathered is only viewable while physically being at the telescope itself. This is a problem, because the telescope is about 20 miles out of town, which means that viewing the telescope data is a bit of a chore. Another problem, is that the status of the telescope is sent to the lowell observatory, but this status may not actually reflect the physical state of the telescope.

Web Application and Closing Statements

By creating a web application that shows all of the weather data obtained by the telescope, as well as other pieces of data like project queue and the last issued command to the telescope, we should be able to make full use of the telescope without the need to drive to its location. This should make any work that must be done with the telescope much more efficient, as well as mitigate any time loss or telescope damage from dome misalignments or miss, or any other problems that may arise.

Our web application will provide several improvements to the current system including saving telescope users time, mitigating the possibility of telescope damage, and providing the public with the information gathered from the telescope. Not only will our project lend to NASA's effort to catalog more NEOs and reach their goal, it will improve the quality of work that involves the telescope, and perhaps provide additional public support to the project, by allowing the public to directly view the fruits of everyone's labors towards the FRoST project. Happy dance.

References

- Pub. L. No. 109–155, December 30, 2005 (codified at 42 U.S.C. § 16691).