LSC Six-Month Progress Report

Organization Vassar college Gravity Wave Group (VCGWG)

Report Date February 15, 2005

Attachment A / LIGO I

The Vassar College Gravitational Wave Group (VCGWG), consisting of Professor Eric Myers and up to four undergraduate students, was admitted to the LIGO Scientific Collaboration in August 2004. The primary goals of the group during this reporting period, as described in MOU Attachment A (LIGO-M040267-00-M), were to facilitate the development and deployment of Einstein@Home, a Distributed Public Computing project, which will use donated computer time from many thousands of personal computers to analyze LIGO data. Evidence of the success of all involved in this project is provided by the fact that a more or less fully functioning Einstein@Home was opened to the public in mid February.

The main contributions of the Vassar group have been in learning how to use the BOINC software and manage BOINC projects, in writing and improving the screensaver graphics code, and in assisting with merging the graphics thread with the science thread to create a complete BOINC application for Einstein@Home. Thus all of the primary deliverables for the period have been completed. The accomplishments of the group and remaining work are described in more detail below.

Some tasks listed in the MOU were actually performed by others with better background or appropriate skills or available time, and some subsidiary tasks have been either only partly finished or deferred. These are also noted below. One result of the screensaver graphics development is a potentially useful tool for 3D visualization of LIGO data (real or simulated) by mapping it to the surface of a rotating sphere, which can be manipulated via the mouse and keyboard.

The term of the proposed work in the next period is shorter than the usual six months. Eric Myers accepted a tenure-track position at Vassar in the Department of Physics and Astronomy, beginning in July 2002, on an accelerated tenure schedule, which took two years off the clock. This unfortunately resulted in a review for renewal of contract in spring 2004 after only three semesters, and there were some difficulties. As the senior staff of the LIGO Laboratory and the LSC are already aware of, the Vassar administration decided to not review the contract, contrary to the wishes of the department. Myers exercised Vassar's appeal process in the Fall of 2004. The decision of the appeal committee, rendered in late December 2004, failed to overturn the original decision. As a result, Myers will end his association with Vassar at the end of June 2005, and the VCGWG will dissolve at that point. Given this, the proposed activities for the VCGWG in the next period extend only up through June 2005, and only consist of continuing work already started or activities related to the operation and improvement of Einstein@Home.

During the previous six-month period Myers has worked both independently and in collaboration with the Einstein@Home development team, consisting primarily of Professor Bruce Allen and his student, David Hammer, of the University of Wisconsin at Milwaukee (UWM), Bernd Machenschalk, Reinhard Prix, and Steffen Grunewald of the Albert Einstein Institute (AEI), and Teviet Creighton at Caltech.

Undergraduates at Vassar have not contributed directly to the work of the VCGWG during this reporting period, but some have contributed indirectly and their names should be mentioned. Kimberly Lefkowitz worked over the summer of 2004 (immediately before this reporting period) on setting up a test BOINC project at Vassar. She had already graduated from Vassar, and she is now employed by a project called PlanetQuest to set up a BOINC project for them. Luc Peterson is a bright and capable junior at Vassar, skilled in both math and physics and familiar with the fundamentals of GR at the level of the introductory text by B.~Schutz. Luc has done some background work and library research for Myers in preparation for calculations to estimate the amplitude of gravitational waves generated by intercommuting cosmic strings. Marco Almeida is a capable physics student with strong programming skills. He has also done some background research for Myers, and in the fall of 2004 he re-assembled our small 8-node Beowulf cluster when it was taken out of storage after laboratory renovations made the previous summer. Indra Shottland and Matt Rosenfeld have recently been assisting in setting up a 4-processor server at Vassar with over half a Terabyte of RAID storage.

Accomplishments during the Period

It may seem to reviewers that creating "cool" screensaver graphics is not a direct contribution to the science goals of the LSC. It has certainly seemed so at times to this author. The argument that it is in fact a useful contribution should therefore be made explicit. Einstein@Home has the potential to provide the LSC with the massive amounts of computational power needed to process LIGO data to the limits of the detectors at their design sensitivity. The "cooler" the screensaver, the more people will join Einstein@Home. The more users who join Einstein@Home, the more processing power is made available to the LSC. Thus a "cooler" screensaver will provide the LSC with more computing power. Einstein@Home also has great potential for outreach to publicize the scientific goals and accomplishments of LIGO, the LSC, and the gravity wave physics community in general.

The specific activities and accomplishments of the VCGWG during the period are listed below, possibly in more detail than the reviewers care for. The added detail may be useful in the event that Myers leaves the project completely. The reviewers are welcome to skim the bullet points to get the big picture of the activities of the period and only read the parts they actually find interesting.

* Screensaver Graphics:

Bruce Allen and David Hammer quickly wrote the original starsphere routine, consisting of stars and constellations from the Hipparcos catalogue (extracted from the open source planetarium program Stellarium). Myers added a list of known pulsars, shown as purple dots. The distribution of the pulsars shows dramatically the band of the Milky Way galaxy across the starsphere, and two clusters are clearly visible in the southern celestial hemisphere corresponding to the Magellenic clouds. Bruce Allen and Graham Woan added a list of known supernovae remnants (SNR's), shown as dark red dots, during the November meeting of the LSC. X-ray sources which are also candidates as GW sources will be added shortly (there are only six at present). Myers restructured the code in several ways, including improvements to the animation event routine (the rendering callback) so that the rotation speed could be manipulated and so that interesting special animation effects can be added more easily in the future.

The screensaver also shows the current search position with an orange marker, which moves around on the starsphere as the search progresses. The sky coordinates of the search position (Right Ascension and Declination) are shown in the lower right corner of the screen. The original search marker consisted of two crossed lines and two concentric circles, but with this configuration it was not possible to see if it was actually aimed at a particular object. The crossed lines where therefore removed from the inner circle, and this functional requirement is the reason the marker bears some resemblance to a gun-sight. The change alerted us to the fact that the Crab pulsar was not in the original list of known pulsars (it was quickly added). The way the search marker was originally drawn was also overly simple, resulting in noticeable distortions when near the celestial poles. When the cause of this problem was finally deduced the code for the search marker was improved to correct for this.

The screensaver also shows the current zenith positions of the LIGO and GEO detectors, projected onto the sky as colored "L" shaped markers. The positions of the markers are updated in real time so that they always show the correct current zenith position, provided that the clock on the client computer is set correctly.

The rotating starsphere is interesting and mesmerizing. It has been made a bit more mesmerizing by adding a very slow up and down wobble of the viewpoint and a slow zoom inward and outward. The code was changed internally to allow arbitrary changes of viewpoint using polar coordinates, which may be useful for more interesting special effects in the future. The time periods for the wobble and zoom are on the order of many rotation periods and are incommensurate. As a result, the user will not see quite the same scene each time the starsphere rotates, but the changes are not noticeable unless one actually looks for it.

The screensaver graphics thread was successfully merged with the LALapps code needed to compute the F-statistic for a particular data segment (SFT) and a given GW template. The validation of the output of this application has been the focus of much of the effort of most of the Einstein@Home development team in the past several months. A more detailed procedure was developed to test the graphics code and then merge it in with the science code is described later.

* Mouse and Keyboard Interaction:

The graphics thread of a BOINC application is primarily intended to be show as a screensaver, but it is also possible for users to view the graphics in real time in a separate window as the application runs. This allows for some user interaction via the mouse and keyboard. The mouse manipulation routines are already in the production version of the current BOINC application. Users can zoom the scene in or out by holding down the right mouse button and moving the mouse up or down. It is even possible to zoom inside the starsphere, which provides an interesting alternative view. The starsphere can be rotated manually by holding down the left mouse button and moving the mouse up or down or left or right. Limits have been placed on how far the user can zoom in our out or rotate the sphere, to avoid a state where the starsphere cannot be viewed. The changes the user makes in the window view of the graphics thread propagates to the screensaver (it's the same thread just displayed in a larger window) but are not (yet) persistent beyond the end of a particular workunit.

Keyboard commands have also been added for user interaction. These are functioning in the test branch of the code but are not yet in the production application. Most keys simply turn on or off the display of a particular part of the scene -- the constellations, the pulsars, the SNR's, the stars, and the static parts of the display. It turns out that BOINC and GLUT handle keyboard events somewhat differently on Unix and Windows. On Windows the keyboard callback function receives the keycode of the *key* the user has pressed, while on Unix the callback receives the character code of the *character* corresponding to they key pressed. This means that control characters and shifted characters have to be handled slightly differently on the two platforms. The code has been written so that it works on both platforms without having to write two separate routines. In the process it was discovered that the Unix implementation of graphics in BOINC does not yet invoke the published API callbacks. This should be relatively easy to add, and the keyboard will then work for the Einstein@Home application on Linux.

All of the keyboard functions for changing the Einstein@Home graphics must currently be invoked with the control key held down. This is in anticipation of a feature to be requested for the BOINC client to allow users to manipulate the graphics even in screensaver mode. Ideally a screensaver should simply get out of the way whenever a user presses a key or moves the mouse, but some latitude is possible. For example, on MacOS X the screensaver actually continues if the mouse is moved. The idea there is probably that it is common to move the mouse out of the way without intending to start using the computer. The screensaver still exits when a key o mouse button are pressed. Similarly, the Einstein@Home screensaver would be even more interesting if users could manipulate the graphics in screensaver mode, but this should only be possible for a limited set of keys. On windows it might be desirable to have a help menu pop up if the user presses the F1 key. More generally, it should be possible to make the graphics respond to commands if a control key is held down, but the screensaver should bow out if any other key is pressed, perhaps even a control key which does not have a function bound to it. This idea needs to be proposed to the BOINC developers community and discussed in more detail before it is implemented, if it is even accepted at all, but the requirement of holding down a control key has already been implemented in anticipation of this addition. The benefit is that it will make the Einstein@Home screensaver even more interesting and therefore draw even more users who will donate their computer time to the project.

* Animated LIGO Logo:

Myers has created an animated version of the LIGO logo, which could be included in the screensaver. Concentric circular arcs propagate out from the upper left corner of the screen and fade into the dark background. While the design for this was independent, it's clearly a fairly obvious idea; it was later learned that Teviet Creighton had already created an animated GIF image along the same lines for a prototype of the screensaver graphics.

Our colleagues at GEO may object to the LIGO-specific reference in the screensave, but it would be possible to add a GEO reference as an animated "L" in the lower left corner which undergoes occasional or continuous quadropole oscillations.

These features are candidates for inclusion in the screensaver but are not a part of the current production code. It's been useful to play with several ideas and then keep only the best, or to arrange to let users decide which of many optional items they prefer to view.

* Bitmapped "Decorations":

In addition to the rotating starsphere the screensaver has some printed information about the user, the host computer, and the search in progress. The screen is also decorated with the label "Einstein@Home", and below this the phrase "World Year of Physics 2005". Unfortunately the size of the available fonts from GLUT is really too small for this. GLUT supports two basic kinds of fonts, bitmapped fonts and stroke fonts. The bitmapped fonts are nicer, but only available in limited sizes (up to 24pt). Stroke fonts, can be drawn in any size, but have a crude line-like appearance. There is therefore a need to put something nicer on the screen in place of the "Einstein@Home" label.

This has been addressed by using OpenGL's "texture mapping" feature, where a bitmapped image is mapped to a polygonal region of the screen. BOINC already includes a C++ class for texture maps, and this has been used to put a rectangular logo for the World Year of Physics on the screen in the lower right corner. But the available tools in the BOINC software only allow mapping the texture to a rectangular region. Myers has therefore written a more general class method to map a texture from an arbitrary polygonal path in a bitmap image file, and used this to map a texture to the corner with a cutout for the starsphere. This has been demonstrated and tested in the test branch of the code by putting images from a draft APS poster in the upper corners of the screen. The upper left corner shows the head of Albert Einstein looking down on the starsphere, while the upper right has the phrases "Catch a wave from space" and "Einstein@Home" in larger more professional fonts. These graphics are still crude, so we are asking the APS to provide us with nicer graphics, which fade naturally to the background screen color (see below).

* Project Specific Preferences:

BOINC provides mechanisms for users to specify both project independent preferences (mainly dealing with how much work to accept, how much disk space to allocate) and project-specific preferences. For example, for the SETI@Home screensaver users can select an overall color scheme of either 'Desert Sands' or 'Tahiti Sunrise'. Myers has learned how to implement project specific preferences for Einstein@Home and demonstrated this by adding an item for controlling the display of static graphics. Some users are concerned that the static portions of the screensaver will "burn" into their screens. Most modern computer screens are immune to burn-in, but this is still a user concern, which we can address. On the server side we simply change one PHP file, for which the BOINC team has already provided a template. On the client side Myers has written the code to parse and implement this preference and turn off the display of the static parts of the graphics. This has been tested on the Pirates@Home test project, and on the "albert" test server, and the client code is a part of the test branch, but this has not yet been incorporated into the production application or server.

It is straightforward to add new project-specific preferences to the project should the need arise. Examples are provided in the server side and client side code for adding check-box items, pull-down choices, and integer and floating point numerical inputs, but these are commented out for now.

* Discussion Forums:

BOINC provides a mechanism for creating and maintaining discussion forums for our users to discuss various aspects of the project and the applications, both amongst themselves and with the project administrators and application developers. The feedback from beta test users has been very useful during the commissioning of Einstein@Home. At present only one discussion forum and one "help-desk" forum have been created. Myers has studied the traffic in these discussions as well as the forum sections implemented by other BOINC projects, and he has designed a set of separate discussion forums for the Einstein@Home project. These have been set up on the "albert" test project for comment, but not yet implemented on Einstein@Home. Doing so only requires running one PHP script from the command line at the appropriate time.

* Apple Installation Process for BOINC:

Myers has assisted David Anderson of Berkeley, the project director for BOINC, in creating an installation process for the Apple platform. Part of this is long term planning for an Apple "application bundle" that fits in with the MacOS X GUI and Framework. When this is finished Apple users will be able to install BOINC on their computers the way they install many applications, by simply downloading and opening a disk image and then dragging and dropping a single item to their Applications folder.

But we are not there yet. Myers has therefore written an AppleScript program to install BOINC on a Mac as a "personal" installation. Users download a disk image containing this script and a prototype BOINC folder. When they double click on the installation script it installs BOINC in the user's Library folder and performs an initial dialogue, which attaches the user to a BOINC project and optionally starts BOINC there and then and can arrange to start BOINC at login.

This has been tested on Pirates@Home and may become part of the distribution process for the next version of BOINC on Mac. This is general to BOINC and is a contribution from Einstein@Home to BOINC, but it will also help Einstein@Home by making it easier for Apple users to join Einstein@Home. Future improvements to the BOINC "manager" should render the AppleScript installer unnecessary.

* Test BOINC Applications:

In the course of developing the Einstein@Home graphics thread and working on testing BOINC code and management of BOINC projects Myers has written several independent BOINC applications. The available BOINC documentation was somewhat sparse (as can be expected for a new and rapidly developing software project) and the examples were therefore not instructive. For his own education Myers wrote the simplest "hello world" program for BOINC, and made that available to the BOINC community along with documentation. This has been useful to new BOINC programmers and will be useful for a programmer who joins the work in adding to or maintaining Einstein@Home. Myers then wrote the simplest graphics program for BOINC. This just turns the screen yellow, so it was called the "yellow, world" program. The next program in the series exercises the full BOINC control API and graphics API, so it was named "lalanne" after exercise guru Jack LaLanne. These programs have remained very useful for testing whenever significant changes are made to the BOINC software, which happens continually at irregular intervals.

At the request of Christian Beer, a programmer who has contributed to BOINC testing and to improvements to the PHP code for BOINC projects of particular use to Einstein@Home, Myers wrote a BOINC application program which generates a small amount of credit and then exits with a randomly chosen error code (zero either included or not). This application is called "roulette" because of the similarities it bears to "Sicillian" roulette, which is like Russian roulette but with a bullet in all chambers save one.

Myers also wrote the 'scroll' application, which takes text from a file and displays it on the screen so that it scrolls upward and off into the distance while a field of stars approaches, much like the opening credits in the Star Wars movies. This was a test of GLUT fonts, and was also used to test the arbitrary URL download work described below.

* Pirates@Home:

In the summer before this reporting period Myers and Lefkowitz set up a test BOINC project called Pirates@Home, and this has continued to be of use in several ways. Most directly it has been used for testing the newest graphics code as it is developed. Indirectly this project has served as a test of the BOINC software and project management and upgrade procedures. It has also been used for testing web design, forum design, and testing the Apple installer for BOINC, without interfering with the development of the Einstein@Home site.

Pirates@Home presently has about 1200 active users and at least double that many active hosts, primarily for the windows platform. Most if not all users are now participating actively in Einstein@Home and only divert their processing power to the test project when occasional tests are released. Pirates@Home therefore contributes to code development but is not taking away any processing power from current production activities. Pirates@Home will be decommissioned in June of 2005 before Myers leaves Vassar.

* Einstein@Home Website and other Test Sites:

Other BOINC projects were set up to test Einstein@Home code and project software, including the "albert" site (at albert.phys.uwm.edu) and the "einstein-test" site. Myers contributed to the configuration of the albert project and used it for testing web design and forum design. Allen, Hammer and Myers performed a complete upgrade of the BOINC software for the albert site during the November LSC meeting. The albert site was then used as a template for the creation of the production Einstein@Home site.

With contributions from Bruce Allen, Myers wrote a web page describing the many elements of the starsphere screensaver and documenting the mouse and keyboard commands for manipulating the graphics.

* BOINC Download from an Arbitrary URL:

The work flow for processing Einstein@Home workunits requires several large SFT files. BOINC is set up so that data files are all downloaded from one central server, but it would be useful if these data files could be downloaded from an arbitrary URL, such as references to one or more LDR repositories. Myers has modified the appropriate BOINC code to allow a data file to be referenced by an arbitrary URL included in the workunit template. This requires that the size of the file and an MD5 hash of the file also be provided to verify the integrity and authenticity of the downloaded file, but this information is already available in LDR.

The modified code was tested using the 'scroll' application on Pirates@Home by having the BOINC client download the complete text of "Treasure Island" directly from the Project Guttenburg repository of copyright-free texts, which was then displayed "Star Wars" fashion in the screensaver (which takes almost six hours). Additional changes are planned to allow more than one URL to be specified for any data file. The BOINC client can already accept multiple URL's for a file and will try each in order until the file has been properly downloaded, which allows for redundancy and load balancing for the data sources.

This mechanism for specifying an arbitrary file URL for download is different from the time-zone based load balancing scheme created for BOINC by Bruce Allen and presently in use on Einstein@Home. In that scheme, which is very clever, the time-zone offset of the client, which is known to the project scheduling software, is used to select a download site, either at UW Milwaulkee or at AEI, and then all files are downloaded from the "closer" site. This is sufficient for present operation of Einstein@Home but does not allow downloads from an arbitrary URL.

* Test Programs and Protocol:

In order to develop the graphics thread for the Einstein@Home screensaver while the science thread was also being actively developed a set of several testing stages has been adopted for taking new graphics features and incorporating them into the full BOINC application (both science and graphics threads). This is important in that the graphics thread development should not interfere with the science mission of Einstein@Home. It has meant that new graphics features are not immediately incorporated into the screensaver, but this may actually be good, as the appearance of each new feature can add new interest for users, at least those paying attention to the graphics.

The procedure for adding new graphics features to the main code branch consists of:

1. The graphics are first tested on Linux using a standalone program called the "boinclet". This small program is a test-stand for BOINC graphics, and consists of a simple GLUT main program and a software layer which connects the BOINC graphics callbacks written for the graphics thread to the appropriate GLUT callbacks. This is run on Linux and lets the developer see the graphics as they would appear in a window or as a screensaver.

2. The graphics code is then incorporated into a BOINC application code named 'sextant' and compiled with the BOINC libraries. This is just the graphics thread -- the computation thread of this BOINC application just continuously generates random numbers for a fixed amount of time so as to run up "credit". This program is first run in "standalone" mode on Linux, which means it runs by itself as it would on a client, but not under the control of the BOINC core client. It is then compiled and run in standalone mode on Windows, which catches both BOINC issues between the platforms and graphics differences between the platforms. The code may go back and forth between Linux and Windows as necessary.

3. The sextant application is then run on the test BOINC project Pirates@Home on several hundred client computers, mainly windows machines. The code can also be compiled on Linux as a BOINC application and run in standalone mode, but the executable is not portable to all versions of Linux, probably because of library issues on our build machine (these same problems do not seem to be an issue for the Linux application running now on the production Einstein@Home). Feedback from the Pirates@Home "crew" on bugs, strange behavior, and general comments about the graphics have all been useful. This has uncovered differences between implementations of OpenGL between Windows and Unix and between different versions of Windows. Some of the "crew" inspect the application very closely and try all with the input mechanism, some of them totally unanticipated by the developer (at least at first).

4. The graphics code is then included in a build of the full Einstein@Home application (cfsBOINC and the dynamic library cfsBOINC.so) on Linux. The code being tested is checked into a separate branch named "sextant" in the Einstein@Home CVS to avoid any interference with development and debugging of the science code. Components of the build process (Makefile.in, configure.ac, and the eah_build.sh script) which need to be modified to build on the branch are also checked into CVS on the branch.

   This complete BOINC application, containing both the graphics and science threads, is tested in "standalone" mode using the testStandalone.sh test script.

   Fixes to problems identified in this stage of testing go back to the previous stage for verification. When a particular version of the graphics thread code passes this stage of testing it is tagged in CVS with the version number used in the previous stage on Pirates@Home. For example, the code which ran successfully on Pirates@Home as sextant version 4.69 and which builds on Linux as cfsBOINC and runs successfully in standalone mode is tagged "sextant_4_69" in CVS.

5. The build candidate is then tested in production using the "anonymous platform" mechanism in BOINC. The executable and library are put in place on a Linux machine participating in completion and verified successfully. At this point the code in the side branch is ready to be merged into the main branch.

While the first version of the screensaver graphics has already been merged with the science code without these steps, later code has not yet been through all five steps. Recent changes to the BOINC software have taken us back to stage 2, where our applications build on Linux but not Windows. It is anticipated though that once all stages of testing are completed and the graphics branch is merged into the main branch then subsequent changes will go through the testing stages more quickly.

* APS Liaison:

Eric Myers has assisted the American Physical Society (APS) in publicizing and promoting Einstein@Home by keeping the APS apprised of progress. Myers also met with APS personnel to arrange for the APS to provide graphic design and web design services. A graphics designer named Anthony Foronda has been assigned to create more professional images for the static decorations for the screensaver and possibly for the website. Joanne Fincham, Internet Services Supervisor for the APS, is working on website design for the project web pages.

* 3D graphics visualization tool: map2sph

The ability of OpenGL graphics to map an arbitrary graphic image onto a 3D surface ("texture mapping") suggests a possible use for visualization of LIGO data, either real or simulated. Right now LIGO data plotted in sky coordinates are mapped onto 2D plots, which LIGO scientists have learned to easily read. The same data could be mapped to the surface of a rotating sphere, which could be manipulated using the same mouse and keyboard interface used for the Einstein@Home screensaver. This might be useful for showing simulated events (or someday real events) to the public, or for an addition to the screensaver graphics, or might even be useful for visualization for LIGO scientists. It would probably be most useful as a science tool if it could extract the image directly from the existing 2D plots currently in use rather than requiring the creation of separate images.

A prototype program called 'map2sph' already works on Linux and can take an arbitrary JPEG image and map it to a rotating sphere. This has been tested with a world map and with a mosaic from NASA showing the lights of Earth at night from space. Some additional work will be required to separate this code from the Einstein@Home code; right now they are joined like siamese twins. Even so, keeping some of the code in common is probably useful for ease of maintenance and for easily applying this functionality in either context.

* Graphics Improvements:

The Einstein@Home screensaver now in production has "cool" graphics and is sufficient for public deployment of the project, but a number of improvements have been considered or already designed. Implementing some or all of these will be part of the proposed MOU for the next period. Progress and planning have been detailed in the file Workplan.txt in the Einstein@Home CVS in CFS/screensaver. The primary additions or improvements being considered are:

* LOGO cube: a rotating cube where the faces are texture maps of the logos of participating organizations (LIGO, LSC, GEO, NSF, APS, and BOINC). The cube would spin and wobble on its axis but could also flip occasionally to display the otherwise hidden face.

* Search marker persistence: patches of the sky already searched are marked with a semi-transparent patch to show the progress of the search.

* Progress bar: graphical indicator of how far the workunit has progressed and how far it has to go. (A numerical indication of the percentage of work finished is already shown).

* Hit parade: graphical indicator(s) showing "hits" above threshold in the current search workunit.

* "Texture mapped" fonts: better fonts where each glyph is a separate texture map. These might easily be borrowed from the Stellarium project, or a new font created by the APS graphics designer. The code for doing this can be borrowed from Stellarium and needs only minor modification to work with BOINC.

* Animated GEO logo: a red "L" in the lower left corner undergoing continuous or occasional quadrupole distortion.

* User/Search info: better 3D display of user info and search info, possibly on a moving 3D clipboard or control panel.

* Save state: save graphical preferences/state between runs, so that any changes the user has made to the graphics view are persistent between workunits.

* Graphical events: occasional noticeable events in the screensaver graphics, which will stimulate or renew user interest. Possible events are (1) suddenly zooming into and then out of the starsphere, (2) occasional sine-gaussian quadrupole distortion of the whole starsphere to simulate the passage of a gravitational wave, or (3) occasional animated flares drawn over the position of a supernova remnant (this was suggested by Michael Landry of LHO).

* Picture show: take a series of pictures from a specified directory and show them one after the other, panning around the pictures and fading between them. This is a feature of the screensaver on MacOS X, and making it available in a way that can be locally customized

might make adoption of Einstein@Home more acceptable for institutions with large public computing clusters (eg. colleges and universities that want control of the images being shown).

One improvement, which has been discussed but not yet designed, is to let users specify a preferred search position in the sky using the preferences mechanism. The scheduling software would then try to take this preference into account. This would not bias the overall search because most users would not specify a preference and work would be allocated as necessary to perform the overall search in any case. This feature has the potential for education and outreach activities by stimulating discussion on the likely sources of gravitational waves.

* Work Completed by others:

The Vassar MOU for the period called for the Vassar group to assist in the creation of the code to "validate" and "assimilate" results returned to the BOINC server by the client applications. The "assimilator" and "validator" code was primarily written by Bruce Allen, and the validator was subsequently improved substantially by Teviet Creighton.

* Work Not Completed

Several items listed in the MOU for the period were not completed. The description of the screensaver in 9a) includes a ticker-tape display showing the values of parameters that have exceeded the threshold for that sky position. As yet no communication of the "hits" found in the science thread is provided to the graphics thread. Item 9d) calls for the development of automated ways to maintain and update a distributed set of data servers for Einstein@Home using LDR. The code allowing a BOINC workunit to download a file from an arbitrary URL is a step toward that goal, but nothing has been done to set up distribution of data for Einstein@Home via LDR. Item 9f) calls for the development of back-end tools to carry out follow-up studies. The project has not yet progressed far enough for that.

* Other Progress

Although it was not formally stated in the MOU, it was hoped that one outcome from Myers' participation in the development of Einstein@Home would be increased familiarity with the data analysis pipeline used by the Periodic Sources search group. That has certainly happened, though it cannot be quantified and there is much more to be learned.