SEGP2 Plugin Group Milestone 6 Report

Total Hours Spend: approx. ??? hours

Executive Summary

Milestone 6 began by reviewing what were the leftover work and issues needed to be addressed. The following were the tasks that had been set by the group at the beginning of the sprint:

- Finish up the 3^{rd} pillar of plugins: deployment.
- Create plugins and apply what was created and set up plugins development into them.
- Front-end plugins union: Finalize and have the plugins framework set.

The first and second items were taken up by Ida, Mohammad, Fil and Ken, while Xiaodong, Keane and Huang Jian were assigned to the last item.

Ida managed to update the `plugins_manager.rb` to support extension point, introduced by Mohammad, during the installation of a daemon plugin. She also managed to add an uninstall method and accessor script to uninstall the a plugin from Earth. Mohammad provided his expertise to assist Ida in getting the implementation correct. Ida had spend ?? hours on updating the scripts, while Mohammad spend ?? hours to lend his expertise to her, while implementing the file metadata as an extension point.

Ken created the `list_plugins` script to enable the administrator to see what plugins had been installed. Also, he unified all the plugins related scripts (`sign_plugin`, `install_plugin` and `uninstall_plugin`) into `earth_plugins`. This root accessor script will call the appropriate script to perform the required plugin task. Ken had spend 4 hours creating the scripts. He also spend 3 hours to help in preliminary integration tasks.

Fil created a daemon feature that redirects the daemon status to the front-end and he spent 80 hours on it.

Xiaodong, Keane and Huang Jian had conceptualized and implemented their versions of GUI plugins framework into functional prototypes.

Xiaodong and Keane's implementation used the Rails plugin generation mechanism to get the plugins installed. They require the user to add specific codes into the controllers to activate the plugins. On the other hand, Huang Jian's implementation used his own way to get the plugins installed, without any code activation in the controller. Both ways are similar, but the difference is only in the ways the framework is implemented. Thus, they have agreed to work togather further to consolidate the ideas, or finalize one one as the final framework, while the other as a proof-of-concept branch.

All three spend ?? hours in bringing the concepts to functional implementations.

Individual Reports

Attached are the individual reports of each member of the group.

Ken S'ng's Milestone 6 Report

Executive Summary

I suggested to improve the way the daemon plugins are to be managed, as I find it is not very organized to have a several scripts to manage the plugins. I have created a main script that consolidates all the plugin management scripts (installation, uninstallation and signing). Besides that, I also have added a script to list all the currently installed plugins, as there is no way to know whether what plugins has been installed prior to the script was written. Besides that I provided my assistance to another subgroup to investigate how integration can be done.

Tasks and Activities Assigned

- Overseeing the progress of the developments of the plugins mechanism development.
  - Plugins deployment retrofitting
    - Description: This task is geared towards retrofitting plugins install scripts from RSP to support the latest additions to the plugins framework.
    - Affected files: `script/list_plugins`, `script/earth_plugins`, `lib/earth_plugins_interface/plugin_manager.rb`
    - Git commits: `(pamalite/earth) 96828da9db7b7c5f19ac67f0b336727052d18b9e`
    - Estimated time taken (planned): 5 hours
    - Estimated time taken (actual): 4 hours
- Helping out on pre-integration investigation and analysis.
  - Description: This is not outlined in the M6 plan. I was engaged to help out on figuring out how to use `git` to merge all the subgroup repositories as a preliminary step before the actual integration begins.
  - Idea/Solution: N/A
  - Affected files: N/A
  - Git commits: *Not relevant as the commits done were experimental.*
  - Estimated time taken (actual): 3 hours

Resource Contributions

For this milestone, I have estimated that I had spend about 12 hours (including 5 hours meeting time) over the spread of 3 weeks. I have budgeted 45 hours (15 hours per week) for this semester. The other 15 hours were spend on my research project.

Looking at the hours I felt that I have had a productive sprint, though I am only using a fraction of my budgeted hours. The rest of the hours were used for other academics commitments. However, I managed to further integrate the group to get the daemon and GUI plugins retrofitted and implemented, respectively. Besides that, though being the main contact of the group, I still get to do some coding to help the group further, and lend my assistance to the other subgroup on the integration preliminaries.

Rooms for Improvement

- Better time estimation needed for paper work and small coding parts. Previously, the budgeted time was huge as there were a lot of things needed to be done. However, it became

a habit to give huge budget, and for this sprint, where the main aim is to prepare for the finishing of the project, only required minimal budget. I should give more time for paper work for this sprint.

Xiaodong Cui's Milestone 6 Report

Executive Summary

During this milestone, I worked with other two MSE students on GUI plugin stuff. We had two different implementation plans, one was to use rails plugin to development Earth GUI plugin, other one was to develop Earth own plugin manager, and then create GUI plugin. So, we divided us into two small team, each team was focused on one plan only. I worked with Kun to implement GUI plugin system based on rails plugin system. I created four GUI plugins, one was used to add a method to add tabs, the other one was based on ticket 66, and another one was to convert existing radial graph to a plugin, and the last one was Mr bogus, which was used to show how a simple plugin system could be created.

Tasks and Activities Assigned

- Research on relevant information for rails plugin
  - Sub-task 1: We decided to use rails plugin system to create GUI plugin system for earth, so we need to know how rails plugin system works, and how we use it to create plugin for earth.
    - Description: Research on how rails plugin system works, such as how to create new method for controllers, models or views, how to generate new files, and how to copy files to the correct location using generator
    - Estimated time taken (planned): 25 hours
    - Estimated time taken (actual): 26 hours (08/09/08-13/09/08)
- GUI plugin development
  - Sub-task 1: Modification on existing plugin created last milestone.
    - Description: Modified the adding a tab plugin which developed last milestone, to make it seperated from the user usage plugin, so it can add a tab and a filter field as well.
    - Affected files: plugins/act_as_add_tab plugin folder
    - git commit 954f8e7da15738d694f9e07a8d9cc353778d48c3
    - Estimated time taken (planned): 5 hours
    - Estimated time taken (actual): 5 hours (14/09/08-14/09/08)
  - Sub-task 2: User usage plugin
    - Description: Generator was used for user usage plugin to generate files. All files needed for the ticket 66 can be generated or copied to the correct locations under earth.
    - Affected files: plugins/user_usage plugin folder
    - git commit 954f8e7da15738d694f9e07a8d9cc353778d48c3
    - Estimated time taken (planned): 15 hour
    - Estimated time taken (actual): 14 hours (15/09/08-17/09/08)
  - Sub-task 3: Convert radial graph views to a plugin
    - Description: The graph plugin is a plugin for users want to see radial graph views, this plugin was developed based on existing earth function radial graph, I changed this function to a plugin.
    - Affected files: plugins/graph plugin folder
    - git commit 954f8e7da15738d694f9e07a8d9cc353778d48c3

    - Estimated time taken (planned): 5 hour
    - Estimated time taken (actual): 2 hours (02/10/08-02/10/08)
  - Sub-task 4: Mr bogus plugin
    - Description: Mr bogus plugin was developed just to help other people to understand how a plugin could be created. It has only one controller file and one view file.
    - Affected files: plugins/mr_bogus plugin folder
    - git commit 954f8e7da15738d694f9e07a8d9cc353778d48c3
    - Estimated time taken (planned): 3 hour
    - Estimated time taken (actual): 1 hours (07/10/08-07/10/08)
  - Sub-task 5: Installation and uninstallation scripts, rake files
    - Description: I created a installation and uninstallation script for user to install and uninstall plugin conveniently, also I created rake files for user to generate a gem package for a given plugin.
    - Estimated time taken (planned): 15 hour
    - Estimated time taken (actual): 17 hours (01/10/08-07/10/08)

Resource Contributions

For this milestone, I have estimated that I had spend about 73 hours (including 5 hours meeting time) over the spread of 3 weeks. I spent the first week to gather knowledge on rails plugin, and then spent the following weeks to modified the existing plugin system, and used the knowledge to create new plugins for Earth. I felt I have spent my time effectively, since I spent enough time on research, so it made me more easier in the plugin development stage.

Rooms for Improvement

- The major problem for this milestone was that we could not communicate with the other team member. We agreed to develop the GUI plugin system in different ways, however I also expected the possibilities to combine the two methods together to make a better way for GUI plugin, so I asked help from other team member to get the idea of how to modify the existing files, because our current method can not modify existing files. But due to some reasons, I could not manage to get any help, I hope we can share our code between team members in the next milestone, that will benefit everybody.

Filimoni Milestone 6 Report

Executive Summary

I had proposed to produce a daemon feature in the form of a plugin to relay the status information from the daemon to the web-based GUI of the Earth application. The motivation for such feature arose from the fact the existing implementation of the GUI provides very little clue to the user about the operation of the daemon. This lack of information become even more crucial if the daemon and the GUI are running on different physical machines, in which case, erroneous assumptions about the daemon operation could effectively limit the reliability of the Earth application. However, further investigation into the actual implementation revealed that such an elaborate plugin approach would introduce unnecessary redesign and complexity to the codebase, all in return for the trivial benefits of providing daemon status information through the web-based GUI. Instead, a more simplistic approach of attaining the same benefit was pursued and implemented in this Milestone sprint. The approach uses the underlying (Linux) file system to relay the relevant daemon status information to the GUI while still maintaining the design separation between the GUI and the daemon.

Tasks and Activities Assigned

- Design and Implement Display of Daemon Status Information on Web-Based GUI.
  - Description: This task involved capturing the daemon status information and displaying it on the web-based GUI in real-time.
  - Idea/Solution: N/A
  - Affected Files:
    - app/controllers/servers_controller.rb
    - app/helpers/servers_helper.rb
    - app/models/earth/server.rb
    - app/views/servers/configure.haml
    - script/earthd
  - Git commits: `(ssurfer/earth.git) 563eb803cfbb8ffb687c86b4c473b718a03dffbd`
  - Estimated time taken (planned): 70 hours
  - Estimated time taken (actual): 80 hours

Resource Contributions

At the beginning of this Milestone, some time was spent identifying which particular feature or aspect of the application to be transformed into a plugin-enabled feature. Aside from deciding on the granularity levels of the plugins, some considerations about the relevancy and appropriateness of the plugins to the primary purpose of the Earth application also had to be made. This part of the task initially appeared to be straight-forward and subsequent efforts were made to quickly transform the implementation of some existing features into the relevant plugin forms. Unfortunately, the retrofitting of the daemon to incorporate plugins using extension points had not been fully completed at this point. As a result, the plugin conversion task had to be put on hold until the relevant plugin framework codes were committed.

In the meantime, the development focus was altered slightly to accommodate these delays and one such change involves the relaying of data between the daemon and the web-based GUI. Presently, the only existing mechanism that is able to facilitate any communication between the

daemon and the GUI is the underlying PostgreSQL database. The daemon updates the database based on the changes in the monitored file system and the Rails application retrieves this information from the database for further processing and subsequent display on the browser. This works flawlessly for the file monitoring operations of the daemon, with the GUI passively updating the results when a web request from the application user is received. Unfortunately, this passivity of information relay may not be adequate or suit certain types of operation such as the real-time monitoring of the daemon status. A more direct mean of communication between the daemon and the GUI is more suited for such requirement.

To this end, a more simplistic solution was implemented which involved the periodic updates of a newly created 'daemon status' log file by the daemon. The corresponding GUI front end then simply retrieves the daemon status information from the said log file and displays it on the browser. Alternately, this approach could also be satisfied using the database instead of the log file. However, the approach employing the log file was adopted during this Milestone to quickly complete the proof-of-concept stage in the investigation of this particular design approach and eliminate the inherent complexities that normally arise with the alteration of the database schema. With the successful implementation and demonstration of the proof-of-concept stage results, the feature can then be implemented using the database instead of the log file. This change is necessary to ensure that this particular application feature can operate in a range of situation such as when the daemon and the web-based GUI are running on two different machines. This later expansion to a database-based implementation of the daemon status information update on the GUI is planned for the next development sprint.

Rooms for Improvement

The allocated time of 70 hours which had been initially earmarked for this particular task was in hindsight a bit optimistic. During the planning phase for the current development sprint, it was wrongly assumed that there was some simple way that the daemon could directly relay its status information to the GUI, which would not necessarily involve an intermediate facility such as the log file or the database. The subsequent investigation which revealed that no such simple provision existed also indicated that the incorporation of such feature through direct code manipulation or the creation of an appropriate plugin would take more time than budgeted. This experience clearly highlighted the costly and adverse effect of design decisions based on wrong assumptions on the schedule of software development projects.

Kun Zhou's Milestone 6 Report

Executive Summary

In this milestone, following the progress of research about GUI plugin system, we decided to realize it by two different ways: one is using the plugin system of rails; the other one is using the plugin manager created by ourselves. Therefore, on one hand, I and Cui focused on the first way; on the other hand, huang realized plugin manager. The customer could choose the better way they thought between these two. I changed the ticket about metadata into plugin. Most of the time was spent on the problem about the view part. Now we got a great progress on it.

Tasks and Activities Assigned

- Rails plugin research
  - Sub-task 1: Analyse the rails plugin stuff Cui created in previous milestone
    - On the researh on rails plguin, i have to keep steps with Cui by analyzing the rails plugin stuff which was created by Cui in previous milestone. That is about user usage. It covered database and some front end. So that it is the best way to master the method of creating rails plugin.
    - Estimated time taken (planned): 10 hours
    - Estimated time taken (actual): 14 hours
  - Sub-task 2: Follow the tutorial on website and research on rails plugin development
    - After analyzing the created plugin, I found that is not enough for solving new problem and realization of new requirement. Therefore, I searched on the internet and got some material, which are papers about rails plugin development.
    - Estimated time taken (planned): 15 hours
    - Estimated time taken (actual): 20 hours
- Create new plugin for metadata
  - Sub-task1: Create files about controller, view and model
    - It is easy for me by following the guider from internet.
    - Estimated time taken (planned): 5 hours
    - Estimated time taken (actual): 6 hours
  - Sub-task 2: Polish functionality about view part
    - In the previous milestone, for view part, table can be created by new line addition in some particular application file. In the milestone, search field can be created. Additionally, those two can be created automatically without adding new line.
    - Estimated time taken (planned): 18 hours
    - Estimated time taken (actual): 26 hours
- meeting and discussion
  - In this milestone, most of work need teamwork and cooperation. For this reason, we took a lot of time on meeting and discussion. That is the base of progress.
  - Estimated time taken (actual): 10 hour

Resource Contributions

In this Milestone, I spent a lot of time on the analysis of plugin and solution of some new requirement almost 50 hours. Most of the other hours were mainly spending on working with my group member. But after this milestone, I have a whole structure of rails plugin system and understand the way it works.

Rooms for Improvement

The first problem in this milestone for me is my lacking of experience of development about earth daemon. Fortunately, i have passed and got a lot of skills. It should be improved in the future. Additionally, I cost too much time on solution of a particular problem and tried to gain it individually. I should have sent it to the whole group and discusses with the others. It should be a better way to conquer the wall of difficulties.

Mohammad's Milestone 6 Report

Executive Summary

This report records my activities related to Earth project development for Milestone 6 period from 7-September-2008 to 12-October-2008.
In this sprint,I have finished the following main tasks: review the earth daemon plugin framework, prepare a document about the earth daemon plugin framework design, check the earthd status script and review the plugin installation script.

Tasks and Activities Assigned

- Review the earth daemon plugin framework.
  - Sub Task 1:
    - Description: reviewing the metadata plugins which was created in the previews milestone.
    - Affected Files: `rsp_add_file_metadata.rb, rsp-add-file-metadata.rb, rsp_delete_file_metadata.` `rsp-delete-file-metadata.rb` and `rsp_delete_under_dir_metadata.rb, rsp-delete-under-dir-met`
    - Git Commits: check the commits on the next sub-task.
  - Sub Task 2: Creating a metadata API which can be used for all metadata plugins in future
    - Description: I create an API contains common metadata functionality like: saving, deleting, searching, etc. All metadata plugins should use this API and not use earth models directly.
    - Affected files: `metadata-api.rb`
    - Git Commits: These Commits includes the previous sub-task commits. `mfbDev/earth: 8a1b6113590b28e, 3742a36327dde, 3742a36327dde, 48da91b6f4d, 48da91b6f4d, a4b173b98b7a5, a4b173b98b7a5, 946ba2c55f83, 946ba2c55f83, b715eee7d38, b715eee7d38, cb98d73ac84, cb98d73ac84`
  - Sub Task 3: Testing the metadata API
    - Description: testing the metadata API with the plugins
    - Affected Files: N/A
    - Git Commits: N/A

- Prepare a document about the earth daemon plugin framework design
  - Description: I created a document that includes details about the earth daemon plugins framework design. This document includes the following main sections: Overall structure, extension points, APIs, plugins loading, simple plugin example and steps to create a plugin. So, this document should be a guide or any one who want to create a new earth daemon plugin.
  - Affected Files: `earth_plugin_framework.pdf, earth-plugin-framework.pdf`
  - Git Commits: `pamalite/segp2: 68f83a173bdc, 68f83a173bdc`
- Review the plugin installation script
  - Description: Ida has been working on improving the plugin installation script so it will work with the new framework design (i.e. extension points). My task was reviewing her work and testing it.
  - Affected Files: `plugin_manager.rb, plugin-manager.rb` and `uninstall_plugin, uninstall-plugin`
  - Git Commits: `mfbDev/earth: 465d98fa, 465d98fa, dd4f1ed5a7, dd4f1ed5a7, 0706e307042, 0706e307042, 177e3e4924, 177e3e4924`
- Check the earthd status script
  - Description: We noticed a problem with the (status) script for earth daemon. When we install File-Monitor as a plugin, this script does not work anymore. This took lots of time to figure out the problem. At the beginning, I thought it is something related to Unix sockets since the system hangs on a (socket.receive) method. I did not find any documentation for this method in Rails API or in the Internet!!! But after several attempts, I figure out the problem. Earth daemon is reading the (status) from an instance variable from the File-Monitor. That's why when we install File-Monitor as a plugin, it does not work anymore. The solution for this could be:
    1. rewrite the status script
    2. use File-Monitor as a normal file because it is the heart of earth daemon
    3. create an extension point for reading the status (I am not sure if this is feasible)

    This task is pushed to be finished at the beginning of next milestone
  - Affected Files: `earthd`
  - Git Commits: N/A

Resource Contributions

I estimated that I will spent about 60 hours for this milestone. I have spent about 50 hours including meeting times.

Milestone Observations

I am happy with the outcome from this milestone from our group. Every one is finishing his assigned tasks and we made a good progress. One thing, we need to give more attention to the documentation and code comments on the next milestone.

Yang Qing's Milestone 6 Report

Executive Summary

In this milestone, I was assigned to work on the plugin installation. I spent the first week in understanding the concept of Mohhamod's extension point. In the second week,I was working on upgrading the install script for adapting to Mohhamod's extension point and implementing the uninstall script. In the last week of this sprint,Ken told me that I need to think about the dependancies among plugins. However, due to time is not enough and after discussing with Mohhamod, we decided to complete everything within the first week of milestone 7.

Tasks and Activities Assigned

- Understanding the concept of extension point and understanding how extension points work in earth.
  - Sub-task 1: Have a research on what is extension point.
    - Description: Due to not know the concept of extension point, I read the documentation written by Mohhamod and searched some related information from Internet,so that I can understand what is extension point properly.
    - planed time taken :15 hours
    - actual time taken :15 hours
  - Sub-task 2:Learning about how extension points work in Earth
    - Description: Reading all the code in earth related to extension point so as to understand how the code works
    - planed time taken :15 hours
    - actual time taken :10 hours
  - Upgrading install script ,creating and implementing uninstall script
    - Sub-task 1: Upgrading install script
      - Description:modifing code in the method called install_from _file() in plugin_manager and modifing code in the script of install_plugin to make the installation adept to extension points.
      - planned:10hours
      - actual: 5hours
    - Sub-task 2: implementing uninstall script
      - Description: creating uninstall_plugin.rb in script and a method called uninstall plugin in plugin_manager
      - planned:5hours
      - actual: 5hours
  - communicate with group members
    - Sub-task 1: communicate with group members to see if I did my job properly.

- Description: I attached my code in email and sent to my group members. I asked for suggestions from Ken and Mohhamod. Finally we agreed on that the install script got no problem, but we need to polish the uninstall script. The dependances need to be considered.And discussed about how to implement it.
- Estimated time taken (planed): 20hour
- Estimated time taken (actual): 20 hour

Resource Contributions

For this milestone, I have estimated that I had spend about 70 hours (including meeting hours over the spread of 3 weeks).

Rooms for Improvement

- In this milestone, everything went on very well. The communication had been improved quite a lot compared with last milestone.

Huang Jian's Milestone 6 Report

Executive Summary

In this milestone, I extend "Management", which means it can add new object (file). And then test it with the stuff - use_usage together. These two plugins use the same API in the applicationhelper, and they are different kinds of plugins: One is pluged into the existing files, and the other is created as a new object. Meanwhile, I try to transfer my skill into rails-plugin, because on matter how good and power my technology is, rails-plugin has been mature and tested very well.

Well, during this milestone, I met a big trouble about my laptop, whose motherboard burned twice and take my whole data away. This disaster take me some days to rebuild systems, redo my assignments.

Tasks and Activities Assigned

- Extend Management
- Description: Check the files I'll use, create and/or overwrite. Copy new plugin files into their relative folders.
- Estimated time taken (planned):15 hours
- Estimated time taken (actual): 15 hours
- Test 2 plugins: use_usage and graphic
  - Sub-task 1: Test use_usage
    - Description: Use_usage is a kind of plugin, which can extend existing file. In this milestone, some part of it has been redone.
    - Estimated time taken (planned):1 hour
    - Estimated time taken (actual): 2 hours
  - Sub-task 2: Test graphic
    - Description: Graphic is treated as a new object.
    - Estimated time taken (planned):1 hour
    - Estimated time taken (actual): 1 hours
  - Sub-task 3: Test use_usage and graphic
    - Description: These two plugins share one API in the applicationhelper.
    - Estimated time taken (planned):5 hour
    - Estimated time taken (actual): 40 hours
  - Reason for sub-task 3 using much more hours than I expected:
    - 1: Ruby on Rails has a bug when using "include" statement. The "include" is just read once by rails when the rsp starts, and later, it will not be treated as part of rsp.
    - 2: Therefore, my plugin structure is not as good as for ruby on rails. I need to rewrite API, some part of coding in applicationhelper.
- Chat with the other 2 members about rails plugin.
  - discuss the skill and help them.
  - Estimated time taken (planed): 5 hour
  - Estimated time taken (actual): 5 hour

- Add my technology into rails-plugin.
  - Description: Rails-plugin cannot extending the existing file, which means it will change rsp code very time when adding plugin.
  - Estimated time taken (planed): 10 hours
  - Estimated time taken (actual): 10 hours
  - Summary: In this milestone, I've got not enough time to finish this.
- Build system and redo some work
  - Description: I never thought my laptop's motherboard burned two times in one week.
  - Estimated time taken (planned):10 hours
  - Estimated time taken (actual): 10 hours

Rooms for Improvement

- Talk more: I feel very guilty without discuss with team members. I just try to finish my par mission and the assignment for ABI.
- Backup: How important the backup is. I took more than a week to redo ABI. It's a lesson.