Solar Plant Data Acquisition – Maintenance

Instructions on installing and running the software

Christian Paulino

Teodor Talov

Instructor: Dr. Janusz Zalewski

CEN 4935

Senior Software Engineering Project

Florida Gulf Coast University

10501 FGCU Blvd. S.

Fort Myers, FL 33965-6565

Draft #4

Submission Date: April 30, 2013
1. Introduction

The software, named “daVinci”, retrieves and stores data from Florida Gulf Coast University’s solar plant website SolarEMS.net. daVinci uses browser automation to navigate to the website’s download page. From there it downloads the CSV files provided and stores the contained data into a database. The purpose of doing this is to make the data readily available as well as easier to use the data in a meaningful way. The database also serves as a backup in case something happens to the website.

daVinci is written in the PHP scripting language using a framework called cakePHP. It utilizes the browser automation tool suite Selenium. Because PHP does not directly support Selenium, PHP-Webdriver is used to interface with it. PHP-Webdriver is an API developed by Facebook so that PHP can work with Selenium. The database used is MySQL. Both the daVinci software and the MySQL database are stored on an eBox-4864 Embedded PC. The operating system installed on the eBox-4864 is Ubuntu. Any Linux operating system will work, however, Ubuntu was the one chosen. Apache is the web server installed on the eBox-4864, which is needed to run the daVinci software.

More details on this project are available from a previous report “daVinvi - Data Acquisition from a Solar Plant”.

This write up only provides information how to install and run the data acquisition software on daVinci.
2. Instructions on installing and running the software

These steps can be run on any Linux compatible machine. In case of the CEN4935 class project, most of the steps have already been performed on the daVinci machine. The only step to execute the software is Step 12 (after logging on).

2.1 Installation instructions

1. Install any Linux distribution (Ubuntu preferred)
2. Install Apache 2.2.x
3. Install PHP 5.3 or greater
4. Install MySQL 5.5 or greater
5. Install Java
6. Enable support for PHP and MySQL in Apache
7. Install GIT
8. Install Firefox
9. Navigate to the following directory:
   
   \(/var/www/\)
   
   Clone the following public repository:

   \(\texttt{git@bitbucket.org:solarfgcu/solar2.git}\)

   by using the following command:

   \(\texttt{git clone} \)

   \(\texttt{https://solarfgcu@bitbucket.org/solarfgcu/solar2.git}\)

   BitBucket and Gmail Accounts (same for both services):
Email: fgcusolarplant@gmail.com
Password: fgcu2012

10. Create a database named:

   solar_app

   By executing the following command:

   database create solar_app

11. Install application specific database schema by running the following command:

   php lib/Cake/Console/cake.php schema create

   Answer the prompts with “Y”

2.2 How to run the software. Once the software has been installed, then follow these steps to run it:

1. Navigate to

   /var/www/solar2

2. Start Selenium (in the background) by running the following command:

   java -jar selenium/selenium-server-standalone-2.25.0.jar &

   If you are not sure that Selenium is running, use the top command and look for java thread. Once Selenium starts, it is safe to exit out of it by sending interrupt (pressing Ctrl+C). When Selenium starts the following information should appear on the screen (see Fig. 1);

3. From the same directory, run the following command in order to start importing data from the solar plant:

   php lib/Cake/Console/cake.php import
The sample of executing this statement is shown in Figure 1.

Fig. 1 DaVinci running

Troubleshooting:

1. Error writing to local folders

   The error is caused by inability of the system to write to local folders which are used by CakePHP to store cached data and session information.

   Error:

   `_cake_core_ cache was unable to write 'cake_dev_nb' to File cache

   fopen(c:\cake\app\tmp\cache\models\cake_model_default_media
   ) [function.fopen]: failed to open stream: No error
   [CORE\Cake\Cache\Engine\FileEngine.php

   Resolution: Issue must be resolved by giving the app/tmp folder read and write permissions (777) by running the following command:
2. Unable to connect to Selenium

   This error occurs whenever Selenium hasn’t been started or it has failed to start.

   Error:

   Error: Unable to connect to host 127.0.0.1 on port 7055 after 45000 ms. Firefox console output:

   Resolution: Navigate to the Selenium folder and start it with the following command

   java -jar selenium-server-standalone-2.25.0.jar &

   After Selenium has started, try to import the data again by running the following command:

   php lib/Cake/Console/cake.php import

3. Http status code 412 when using https

   error: The requested URL returned error: 412 while accessing https://{username}@bitbucket.org/{username}/{reponame}.git/info/refs

   fatal: HTTP request failed

   Resolution: This error is due to not having a password set on your account. This can happen if you've created your account using the "Log in with Google|Twitter|Facebook|GitHub" and opted to not create a Bitbucket password during the sign-up process, as shown below. The actual error message you are seeing is a bug,
which is being tracked at https://bitbucket.org/site/master/issue/6018/http-412-error-when-connecting-to.

4. Page cannot be displayed

   This error occurs whenever Apache hasn’t started

   Error:

   Page cannot be displayed

   Resolution: Start Apache by running the following command:

   sudo /etc/init.d/apache2 start

5. CakePHP cannot connect to the database

   If the database does not have the appropriate user, the application will not be able to connect to the database.

   Error:

   Error: SQLSTATE[HY000] [2005] Unknown MySQL server host 'xx.xx.xx.xx:33306' (2) requires a database connection

   Error: Confirm you have created the file:

   app/Config/database.php.

   Resolution: Add the following user to MySQL server by running this commands:

   CREATE USER 'solar'@'localhost' IDENTIFIED BY 'solar';

   GRANT ALL PRIVILEGES ON * . * TO 'solar'@'localhost'