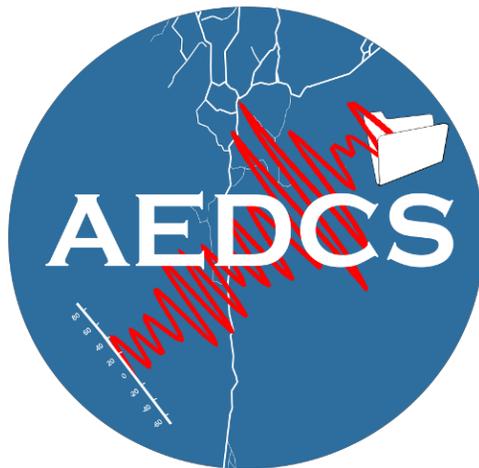


Automated Experiment Data Centralization System User Manual



Version 1.0

NEES and the University of Minnesota

Table of Contents

[Document Versioning](#)

[Introduction](#)

[Audience](#)

[Deployment Options](#)

[Single Machine Configuration](#)

[Client-Server Configuration](#)

[Client Installation](#)

[Windows Client Installation](#)

[System Requirements](#)

[Linux Client Installation](#)

[System Requirements](#)

[Central Server Installation](#)

[Prerequisites](#)

[Linux Central Server Installation](#)

[Installation Steps](#)

[Windows Central Server Installation](#)

[Installation Steps](#)

[System Operation](#)

[Overview](#)

[Configuring an AEDCS Client](#)

[Configure the AEDCS Central Server](#)

[Configuring Pen](#)

[Starting the AEDCS Server](#)

[Starting an AEDCS Client](#)

[Configuring the AEDCS Variables](#)

[Configuring the AEDCS Data Mapping](#)

[Throttling TCP/IP Traffic](#)

[Performing File Verification](#)

[Data Upload Preparation](#)

[Transfer Data from a Client to Central Server](#)

[Stopping Data Transfer a Client to Central Server](#)

[Data Upload](#)

Document Versioning

Date	Author	Change Description	Version
11/27/2013	Michael Boldischar	Initial Version	0.5
9/11/2014	Michael Boldischar	Updated for software changes	1

Introduction

Each NEES site has one or more data acquisition systems. Many of these systems are designed to capture sensor data and store to a local hard drive. The purpose of the Automated Experiment Data Centralization System (AEDCS) is to automatically aggregate data from NEES data acquisition systems. This tool uses a hub and spoke approach to gather data from various sources (See Figure 1). As the data acquisition systems collect data and update the filesystem, the AEDCS detects changes and sends file deltas (changes) to a central server. Also, the AEDCS uses acquisition system specific rules to map the data to the appropriate location in the Project Warehouse directory structure.

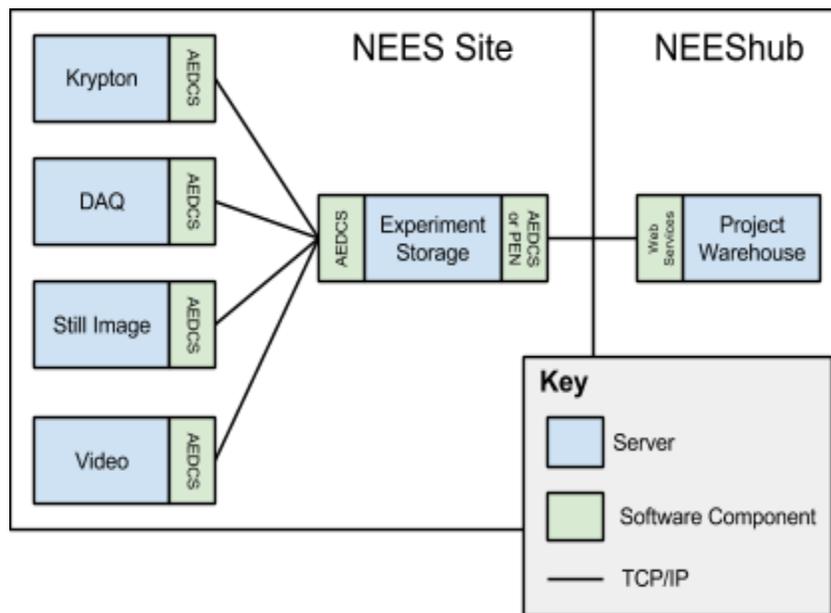


Figure 1: AEDCS Architecture

Audience

This manual is intended for the following users:

- NEES IT Staff
- NEES Researchers

NEES IT Staff is defined as a person who works at a NEES site where NEES experiments are executed. This person is responsible for IT operations and data upload at the NEES site. A NEES researcher is an individual responsible for uploading experiment data to NEESHub. AEDCS software developers are individuals who are interested in compiling and creating custom configurations for AEDCS.

Deployment Options

Single Machine Configuration

In this configuration, a single desktop computer is responsible for gathering data and uploading. This configuration can be used for a small research project where data is gathered and stored on removable media. Also, this configuration may be used if data is gathered offline and uploaded after an experiment is completed. In this configuration, both the AEDCS client and AEDCS server are installed on the same host. See Figure 2 for an example of this configuration.

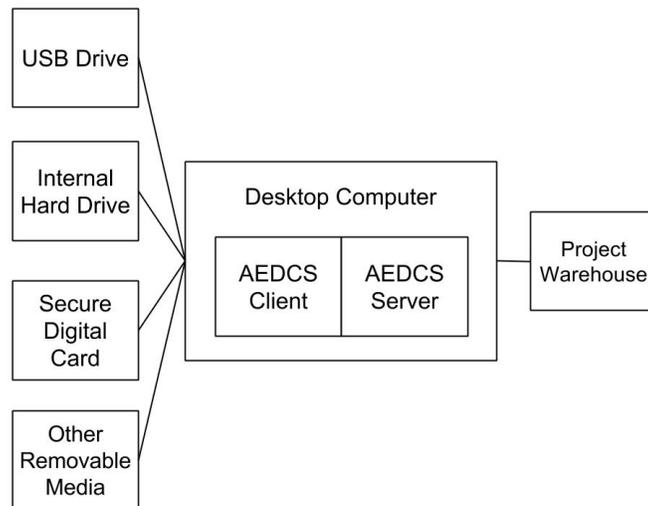


Figure 2: AEDCS running on a single host

Client-Server Configuration

The client-server configuration is the standard deployment configuration for a NEES site. Data may be collected by multiple systems and stored on separate host hard drives. All the data from these separate systems is aggregated by the AEDCS Central Server. From the Central Server, the data is uploaded to NEEShub. See Figure 3 for an example of this configuration.

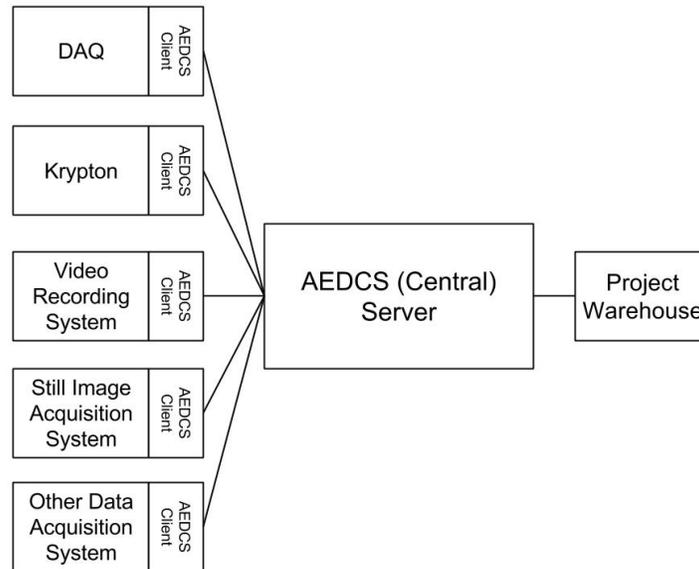


Figure 3: AEDCS running with multiple clients and a single server

Client Installation

Windows Client Installation

System Requirements

- Windows 7 32-bit or 64-bit
- 50MB of free space for AEDCS client software
- TCP/IP network connection to Central Server

Prerequisites

- Installation of [Oracle Java Development Kit \(JDK\) 7u45+](#). Steps to complete this installation are outside the scope of this document.

Linux Client Installation

System Requirements

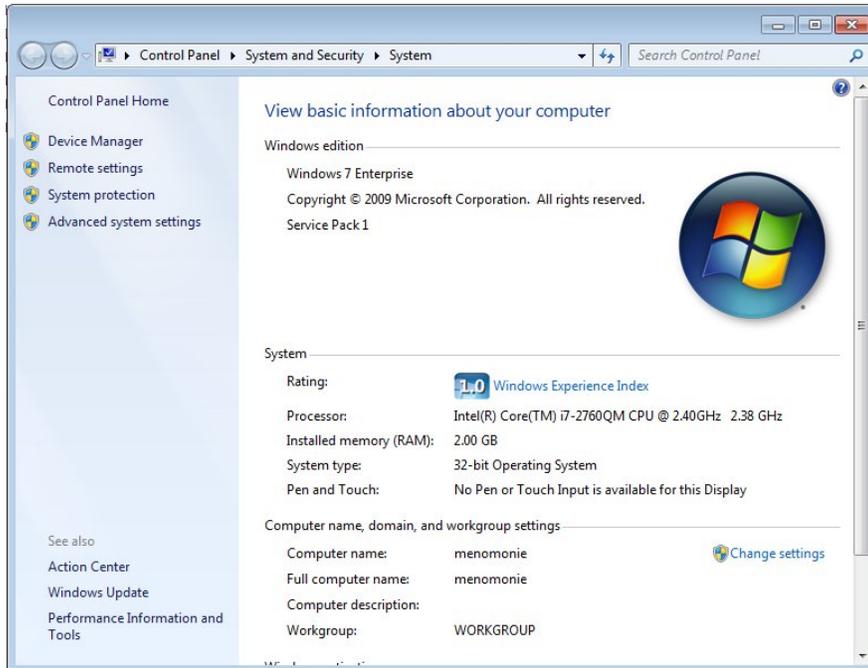
- 50MB of free space for AEDCS client software
- Linux Kernel 3.0+
- TCP/IP network connection to Central Server

Prerequisites

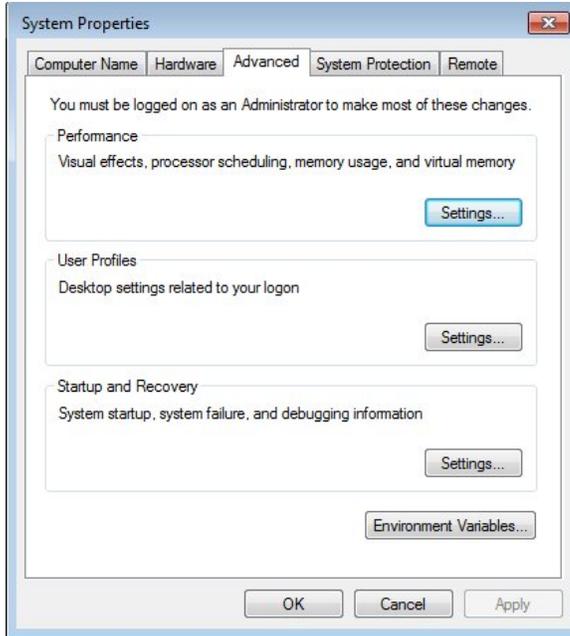
- Installation of [Oracle Java Development Kit \(JDK\) 7u45+](#). Steps to complete this installation are outside the scope of this document.

The following steps are required to properly configure an AEDCS Client.

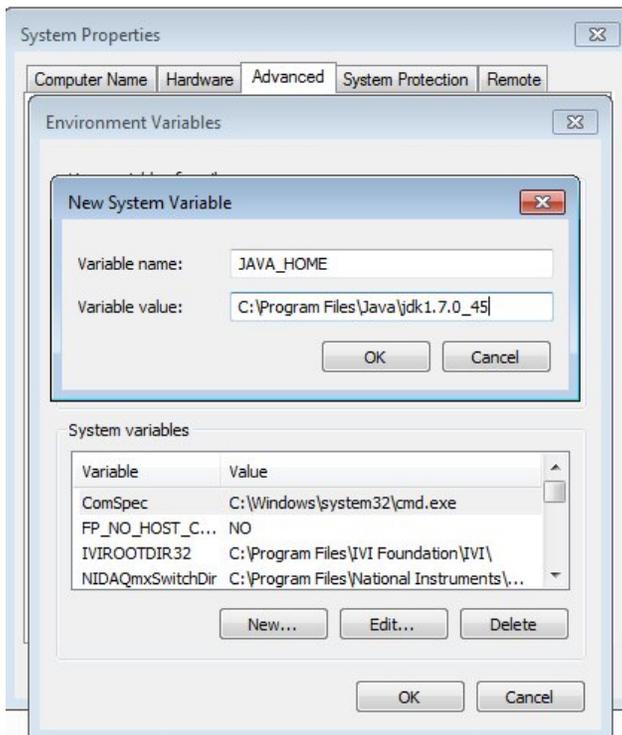
1. Log into the AEDCS client you want to configure.
2. Left click on the Windows “Start Menu” to bring it up.
3. In the “Start Menu”, right click on “Computer” and choose “Properties” from the drop-down menu.
4. In the “Properties” menu, choose “Advanced System Settings”.



5. In the “Advanced” tab of the “System Properties” window, click the “Environment Variables...” button.



6. In the “Environment Variables” window, click on the “New...” button under the “System variables” section to add a variable for “JAVA_HOME”.



7. Set the “Variable name” to “JAVA_HOME”.

8. Set the “Variable value” field to the path to the Oracle JDK directory. For example:
C:\Program Files\Java\jdk1.7.0_45
9. Hit “OK” on the “New System Variable” window to save the new variable.
10. Hit “OK” on the “Environment Variables” window to save the settings.
11. Hit “OK” on the “System Properties” window to save the settings.
12. Navigate to the directory where the AEDCS ZIP file was extracted.
13. Open the file “aedcs.properties” with a text editor (e.g. Notepad++). Each property in this file is on a separate line. Here is an example:

```

1 #####
2 # This is the configuration file for AEDCS clients.
3 #####
4
5 # This is the unique name for the data acquisition system client
6 daq.name=timelapse
7
8 # This is the IP address of the Tomcat server. If this is a single-host
9 # configuration, use localhost or 127.0.0.1 for this value
10 server.ip=192.168.0.117
11
12 # This is the port of Tomcat. The default port is 8080.
13 server.port=8080
14
15 # This is the sleep time for common worker threads in the client
16 thread.sleep time=1000
17
18 # This needs to be set to the maximum POST size for Tomcat
19 # Performance can be increased by increasing the max post size
20 max.byte.size=2097152
21
length: 764 lines: 21 Ln: 10 Col: 11 Sel: 13 UNIX ANSI INS

```

14. Change the value for the property “daq.name” to a unique name for this data acquisition system. For example, a system that gathers data from a timelapse camera might be named “timelapse”. This name will appear on the Central Server website.
15. Change the value for the property “server.ip” to the IP address of the Central Server. For example: “192.168.0.117” (without the quotes).
16. Optional: Update the “max.byte.size” value. Increasing this value will improve transfer speeds. The default value for this is “2097152” (bytes). Here is an example where the

size has been increased to 200MB per request:

```
max.byte.size=209715200
```

Warning: This value must also be changed in your Tomcat configuration before making this modification. Otherwise, errors will occur since Tomcat and the clients will not agree on the maximum POST size for data. This value must match the “maxPostSize” configured in the “Configure the AEDCS Central Server” section.

17. Optional: Change the “server.port” value. This value must correspond with the port Tomcat is running on. The default port for Tomcat is “8080”.

Warning: Changing this value requires a change in the Tomcat configuration, otherwise the client will not be able to connect to the Central Server.

18. Save the “aedcs.properties” file in your text editor.

19. Close your text editor.

Each AEDCS client needs a unique name (e.g. krypton or daq).

Central Server Installation

Prerequisites

Here are the prerequisites for installing the AEDCS Central Server software:

- [Oracle Java Development Kit \(JDK\) 7u45+](#)
- [Tomcat 7.0.47+](#) - Examples in this document assume Tomcat is installed in the following path: “C:\bin\apache-tomcat-7.0.47”
- [NEES Pen](#)
- [Google Chrome](#)

Instructions for installing these software components is outside the scope of this document. There are many resources on the internet that describe how to properly install these components.

Linux Central Server Installation

System Requirements

- Linux Kernel 3.0+
- 50MB free space for AEDCS server software
- Enough free space to keep a cache of all experiment data

Installation Steps

1. Open a terminal.
2. Change to the “root” user.

```
$ su
```

3. Download and install the latest version of the Oracle Java Development Kit (JDK) in the “/opt” directory of Linux. For example:

```
$ cd /opt
```

```
$ tar xzf /home/user/Downloads/jdk-7u51-linux-x64.tar.gz
```

4. Create a symbolic link to the latest directory of the JDK as follows:

```
$ ln -s /opt/jdk1.7.0_51 /opt/jdk
```

5. Download and install the latest Apache Tomcat in the “/opt” directory of Linux. For example:

```
$ cd /opt
```

```
$ tar xzf /home/user/Downloads/apache-tomcat-7.0.50.tar.gz
```

6. Change the ownership of the extracted directory to the user you are running as. For example:

```
$ chown -R <user account> /opt/apache-tomcat-7.0.50
```

Where:

<user account> - the Linux user account you want to run the AEDCS process as

7. Create a symbolic link to the latest Tomcat as follows:

```
$ ln -s /opt/apache-tomcat-7.0.50 /opt/tomcat
```

8. Download the latest stable release of AEDCS from the NEES website to a temporary location (e.g. “/tmp”).

9. Extract the contents of the latest stable release of AEDCS. For example:

```
$ cd /tmp
```

```
$ unzip aedcs-2013.12.02.214.zip
```

10. Navigate to the temporary directory where the ZIP file is located. For example:

```
$ cd /tmp
```

11. Copy the WAR file to your “webapps” Tomcat directory. For example:

```
$ cp /tmp/aedcs/aedcs.war /opt/tomcat/webapps
```

12. Switch to the user account you want to run Tomcat as. For example:

```
$ su <user account>
```

Where:

<user account> - the Linux user account you want to run the AEDCS process as

13. Set the “JAVA_HOME” environment variable to the path where the Oracle JDK is installed. If this variable is not set, Tomcat may not start. For example:

```
$ export JAVA_HOME=/opt/jdk
```

14. Set the “TOMCAT_HOME” environment variable to the symbolic link we created for Tomcat. For example:

```
$ export TOMCAT_HOME=/opt/tomcat
```

15. Start Tomcat as follows:

```
$ cd /opt/tomcat/bin  
$ ./startup.sh
```

16. Open a web browser and connect to the following URL:

```
http://<central server ip>:<port>/aedcs
```

Where:

<central server ip> - the IP address of the host running the AEDCS Central Server

<port> - the port Tomcat is running on

Note: You may need to open a port in the Linux firewall if you are having trouble connecting on step 9.

Windows Central Server Installation

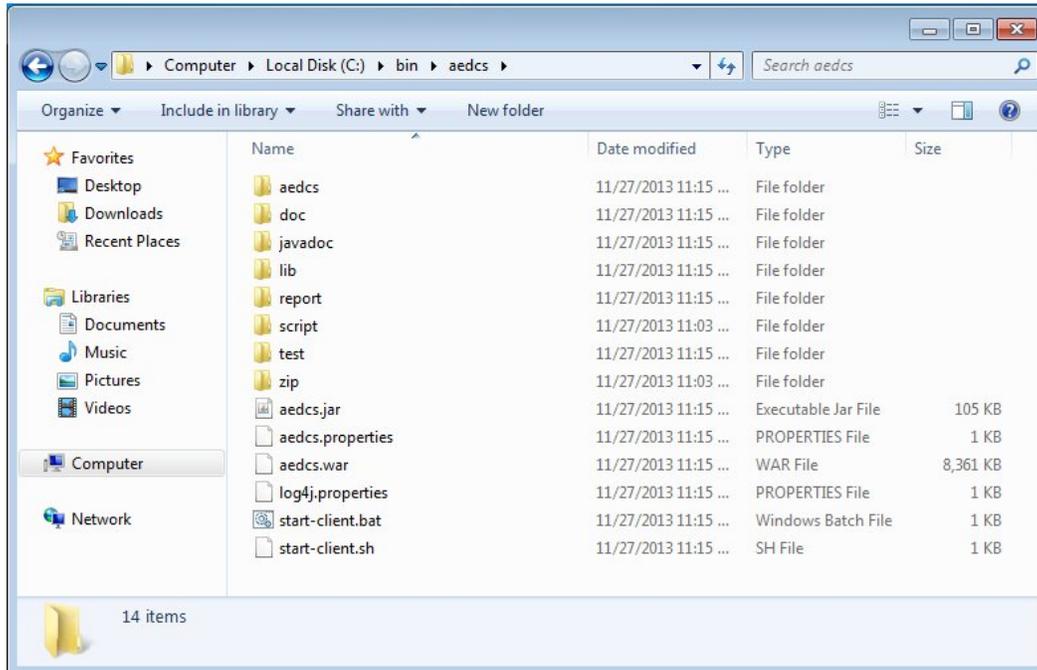
System Requirements

- [Oracle Java Development Kit \(JDK\) 7u45+](#)
- [Tomcat 7.0.47+](#)
- Windows 7 32-bit or 64-bit
- 50MB free space for AEDCS server software
- Enough free space to keep a cache of all experiment data

Note: This guide does not include steps for installing Tomcat or Oracle. There are many resources on the internet which describe how to configure these tools.

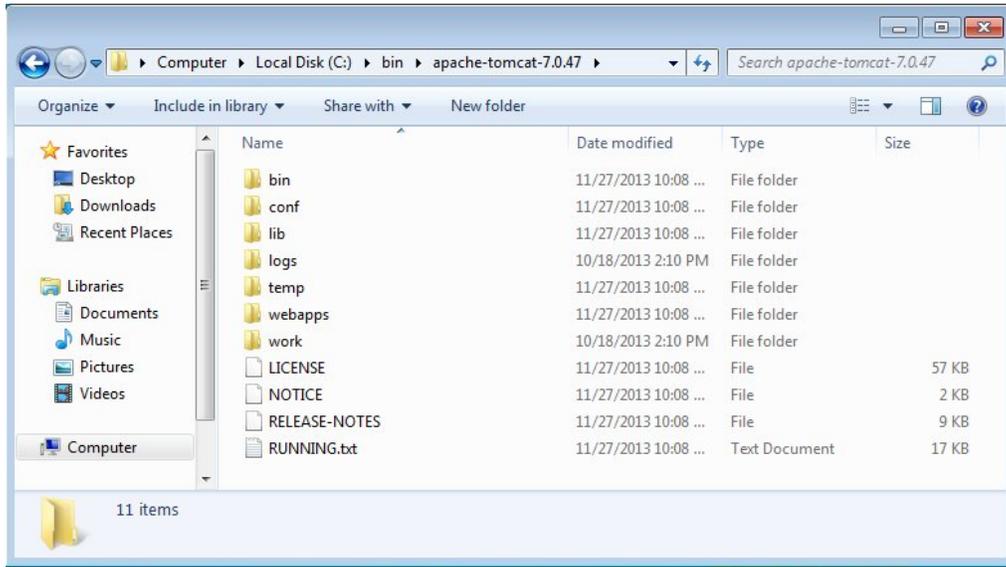
Installation Steps

1. Download the latest stable release of AEDCS from the NEES website to the directory on your computer where you want to install it. (e.g. “C:\bin”).
2. Extract the contents of the AEDCS ZIP file into this directory. The resulting directory should look similar to the following:

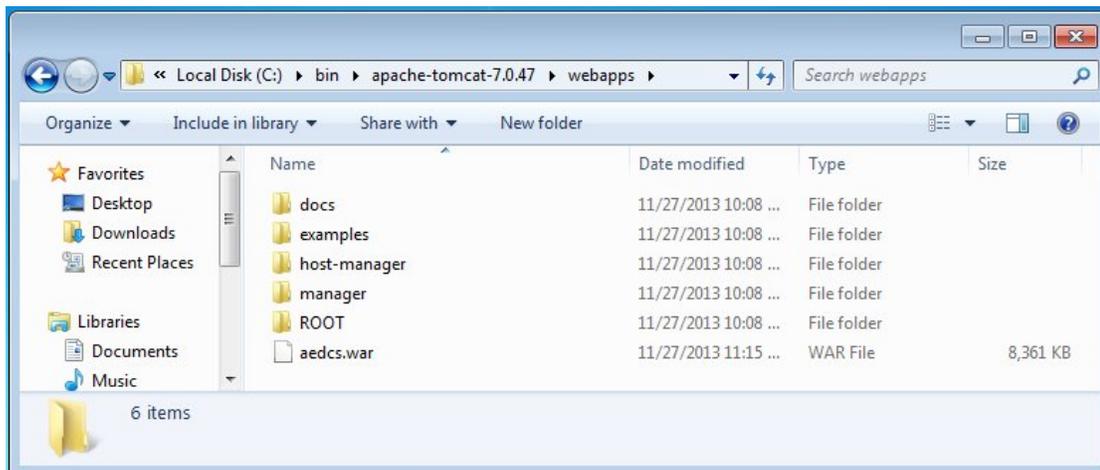


3. Right click on the file called “aedcs.war” and choose “Copy” from the drop-down menu.

4. Navigate to the Tomcat installation directory. It should look similar to the following:



5. Navigate to the “webapps” subdirectory.
6. Right click in the “webapps” directory and choose “Paste” from the drop-down menu. After this operation, the “aedcs.war” file should be copied to the current directory.



System Operation

Overview

AEDCS is designed to perform the following functions during an experiment (See Figure 4):

1. Map acquisition data to PEN - AEDCS has a data mapping configuration for each client. This describes where the data comes from on the client and where it goes on the server.

2. Run on acquisition system - AEDCS can be run on any number of data acquisition systems.
3. Watch filesystem for changes - AEDCS watches specified files on data acquisition systems for changes.
4. Read data from filesystem - Once file changes are detected, the data is read from the client filesystem.
5. Send data deltas to Central Server - The data changes (deltas) are streamed from the client to the Central Server over TCP/IP.
6. Receive data Deltas - The Central Server receives the deltas via Tomcat HTTP requests.
7. Write data deltas to filesystem - The Central Server writes the changes to the filesystem.
8. Verify data integrity - Once data is done being collected, the integrity can be verified by sending the client a "VERIFY" command. This is a good check to make sure the data was uploaded properly and the client data matches the server data.
9. Upload to NEEShub - This feature is currently satisfied by running the Pen tool separately. Pen can be used to upload data to NEEShub from the Central Server.

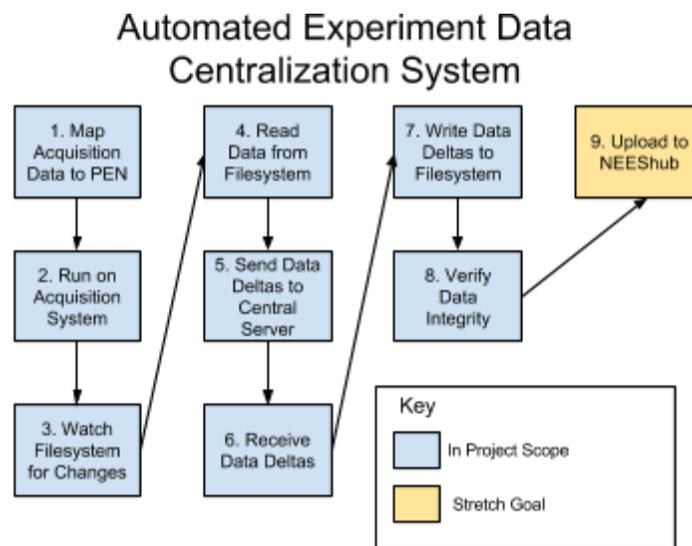


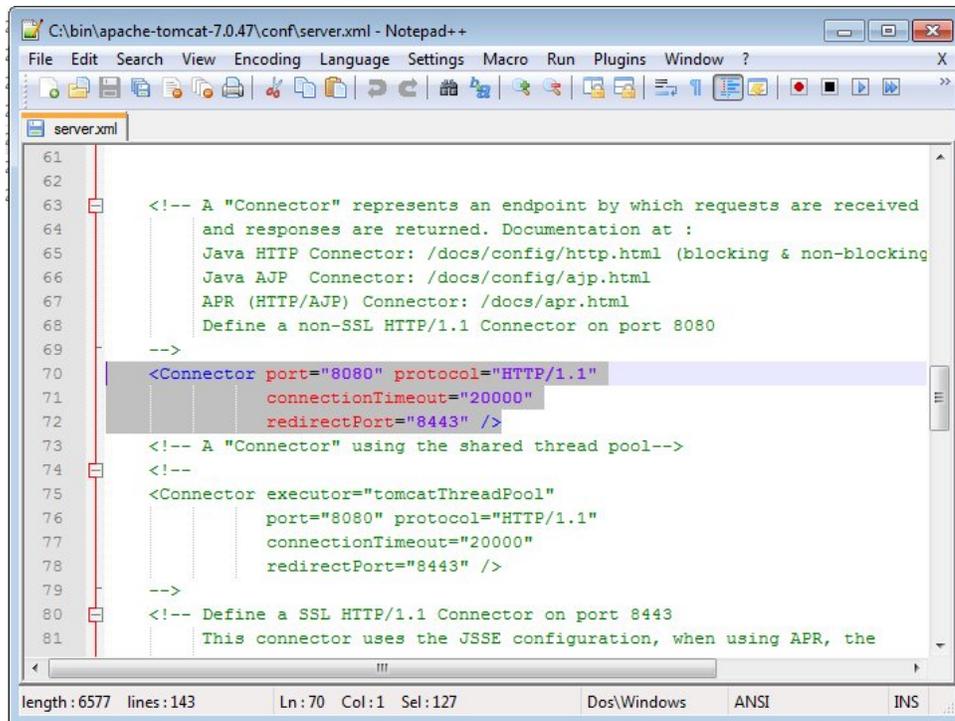
Figure 4: AEDCS Functional Analysis

Configuring an AEDCS Client

Configure the AEDCS Central Server

The steps in this section are optional. These steps are for advanced users only.

1. Open the Tomcat “server.xml” using a text editor. This file is found in the Tomcat “conf” directory.
2. Locate the section of the “server.xml” file with the “Connector” XML element. It should look similar to the following:



```
61
62
63 <!-- A "Connector" represents an endpoint by which requests are received
64 and responses are returned. Documentation at :
65 Java HTTP Connector: /docs/config/http.html (blocking & non-blocking
66 Java AJP Connector: /docs/config/ajp.html
67 APR (HTTP/AJP) Connector: /docs/apr.html
68 Define a non-SSL HTTP/1.1 Connector on port 8080
69 -->
70 <Connector port="8080" protocol="HTTP/1.1"
71 connectionTimeout="20000"
72 redirectPort="8443" />
73 <!-- A "Connector" using the shared thread pool-->
74 <!--
75 <Connector executor="tomcatThreadPool"
76 port="8080" protocol="HTTP/1.1"
77 connectionTimeout="20000"
78 redirectPort="8443" />
79 -->
80 <!-- Define a SSL HTTP/1.1 Connector on port 8443
81 This connector uses the JSSE configuration, when using APR, the
```

3. Set your tomcat connector to the following:

```
<Connector port="8080" protocol="HTTP/1.1"
connectionUploadTimeout="3600000" disableUploadTimeout="false"
connectionTimeout="60000" redirectPort="8443" maxPostSize="0"/>
```

Configuring Pen

The current version of AEDCS leverages the NEES Pen tool to upload data to NEEShub. Pen must be run on the same host as the AEDCS Central Server.

Note: AEDCS integration with the NEES Pen tool is in progress. This will allow automatic upload of data to NEEShub as data is gathered from clients without needing the Pen upload tool.

Starting the AEDCS Server

To start the AEDCS Central Server, follow these steps:

1. Start Tomcat on the AEDCS Central Server by executing standard Tomcat startup scripts located in the Tomcat “bin” directory. For Windows, this is “startup.bat”. For Linux, the startup script is called “startup.sh”.
2. Open Google Chrome on a networked computer, type the following in the address bar, and press the Enter key:

```
http://<central server ip>:<port>/aedcs
```

Where:

<central server ip> - the IP address of the host running the AEDCS Central Server

<port> - the port Tomcat is running on

This will open the Central Server configuration page. If you

Starting an AEDCS Client

The following steps can be used to start an AEDCS client.

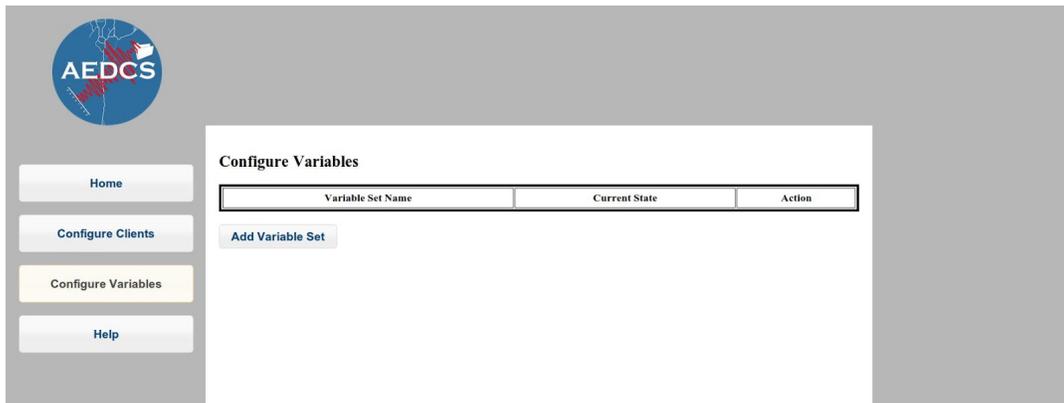
1. Navigate to the directory where the AEDCS ZIP was extracted.
2. Run the “start-client” script corresponding to your current operating system. For Linux, run “start-client.sh”. For Windows, run “start-client.bat”. This will start the client in a shell.

Configuring the AEDCS Variables

Variables are useful in AEDCS for managing paths with elements that change for each experiment, but retain the same structure. Configured variables are global for one Central Server. Variable expansion is performed on each data mapping where the variables are found.

1. Open Google Chrome and navigate to the Central Server website. See “Starting the AEDCS Server” section for more information on this URL.

- Click on the “Configure Variables” button to open the page.



- Click the “Add Variable Set” button to bring up the variable set window.



- Click the “Enable” button to enable this variable set. **Note:** If a variable set is not enabled, the values will not be used during data mapping.
- Enter a value in the “Variable Set Name” field. This should be a unique name that helps identify this set of variables.

Tip: Variable sets can be used to manage variables that change between projects or experiments. Planning can be done for upcoming experiments by creating a new variable set and keeping it disabled until the experiment starts.

- Click the “Add Variable” button to add a new variable entry.
- Add a “Key” and “Value” for the variable.

8. Repeat steps 6-7 for each variable you want to set.

Add New Variable Set✕

Enable

Variable Set Name:

Key	Value
<input style="width: 100%;" type="text" value="site-experiment"/>	<input style="width: 100%;" type="text" value="UMNEXP1"/>
<input style="width: 100%;" type="text" value="site-project"/>	<input style="width: 100%;" type="text" value="UMNPROJ1"/>
<input style="width: 100%;" type="text" value="site-trial"/>	<input style="width: 100%;" type="text" value="Test"/>
<input style="width: 100%;" type="text" value="nees-experiment"/>	<input style="width: 100%;" type="text" value="Experiment-1"/>
<input style="width: 100%;" type="text" value="nees-project"/>	<input style="width: 100%;" type="text" value="8"/>
<input style="width: 100%;" type="text" value="nees-trial"/>	<input style="width: 100%;" type="text" value="Trial-1"/>

Add Variable

Cancel Save

9. Click the “Save” button to save the variable set and close the window and display the result of the operation.



10. Verify that the variable set appears on the list and that the variable set has a “Current State” of “Enabled”. **Note:** Refreshing the browser page may be required to see the change.

Configure Variables

Variable Set Name	Current State	Action
UMNEXPI	Enabled	Edit Delete

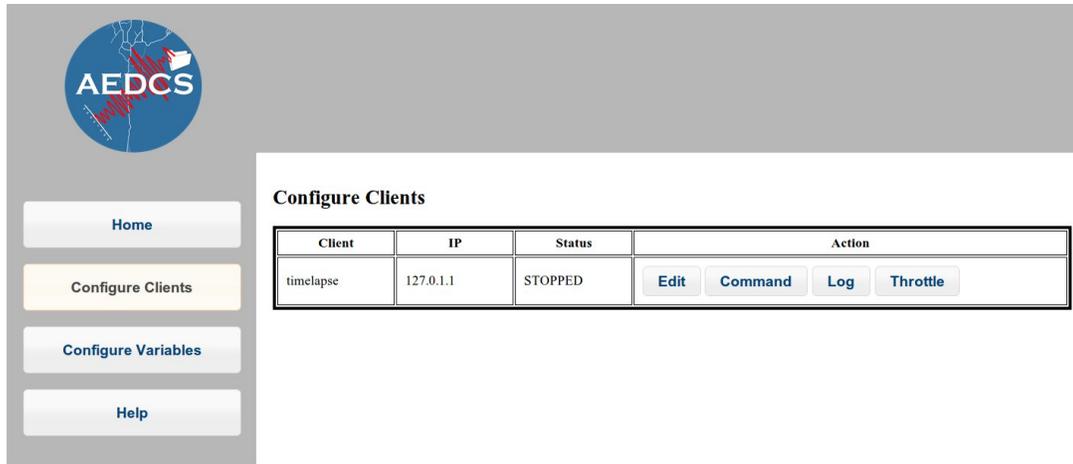
[Add Variable Set](#)

- 11.

Configuring the AEDCS Data Mapping

To configure the data mapping for a client, perform the following steps. **Note:** The client must be connected prior to configuring the data mapping for it.

1. Open Google Chrome and navigate to the Central Server website. See “Starting the AEDCS Server” section for more information on this URL.
2. Click on the “Configure Clients” button to open the page listing all the connected clients. It should look similar to this:



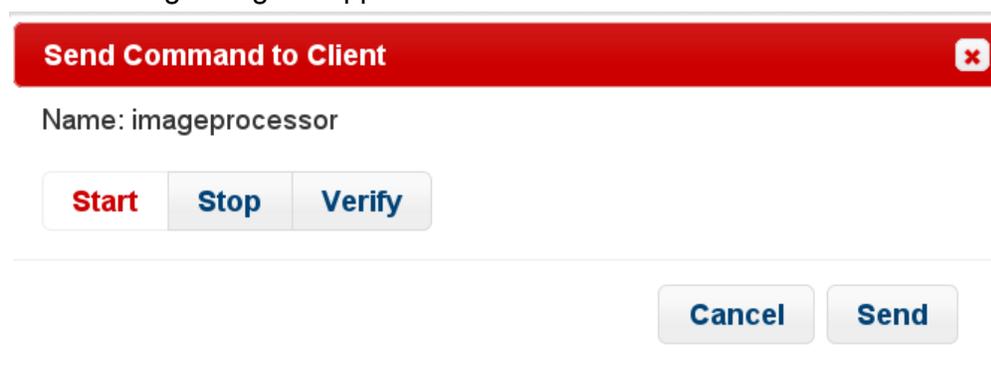
3. Choose the client from the list that you wish to configure. For that client, click the “Edit” button to bring up the data mapping pop-up. It should look similar to this:

Tip: Using variable sets effectively in data mappings can save a lot of work when switching projects and experiments. Develop a common process for saving data (e.g. common directory structure). Then, use variables in data mapping source and destination paths.

Throttling TCP/IP Traffic

Performing File Verification

To stop a client from sending data, click the “Command” button on the “Configure Clients” page. The following dialog will appear:

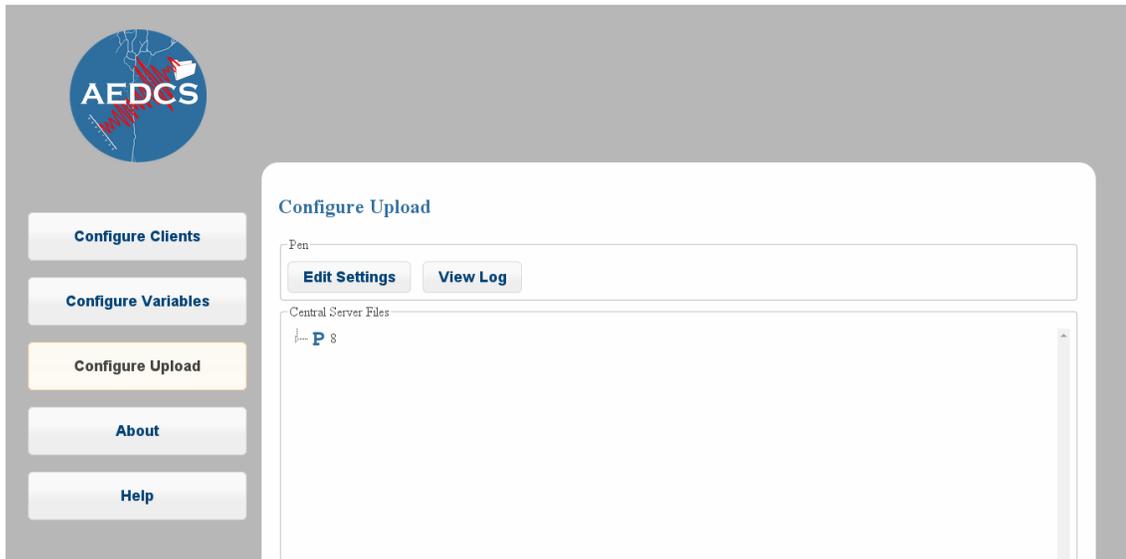


Click the “Verify” button and press “Send” to send a command to perform file verification.

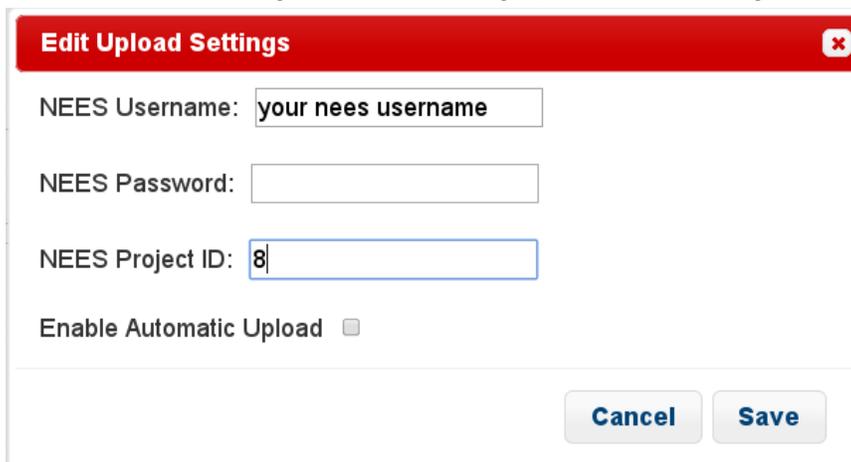
Data Upload Preparation

To start uploading data to nees.org, follow these steps. It is expected that experiment data has

1. Navigate to the “Configure Upload” screen shown below.



2. Click the “Edit Settings” button to bring up the Pen Settings dialog.



3. Enter your nees.org username and password in the “Edit Upload Settings” window.
4. Enter the project ID of the data you want to upload.
5. Decide if you want to enable automatic upload of data as it is gathered from data acquisition systems. If so, check the “Enable Automatic Upload” checkbox.
6. Click the “Save” button to close the window and save your data. Note: the “Central Server Files” area will refresh and show a workspace for the give project ID. It will be empty by default until data is uploaded to the Central Server or downloaded from

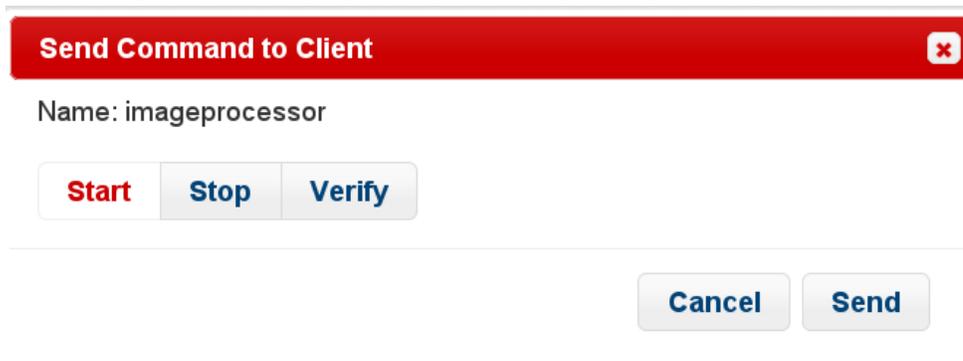
nees.org.

7. Create the directory structure required for your data using the “Create Subdirectory” and “Rename” buttons in the dialog. Now you are ready to start transferring data from the data acquisition systems to the Central Server.

Transfer Data from a Client to Central Server

To start transferring data from a client to the Central Server, follow these steps.

1. To start data transfer from a client to the Central Server, click the “Command” button on the “Configure Clients” page. The following dialog will appear:

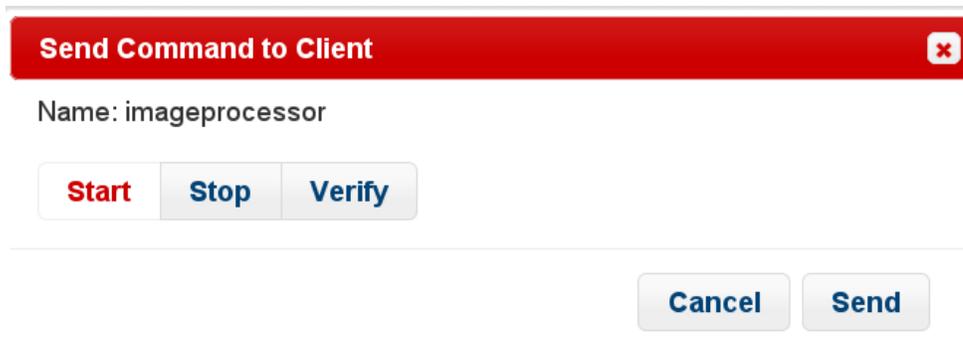


2. Click the “Start” button.
3. Click the “Send” button to send a message to the client. Starting a client will instruct the client software to send data to the Central Server. Once all the data from the client is sent, the client goes into a “synchronized” state.

Stopping Data Transfer a Client to Central Server

To stop a client from sending data, follow these steps.

1. Click the “Command” button on the “Configure Clients” page. The following dialog will appear:

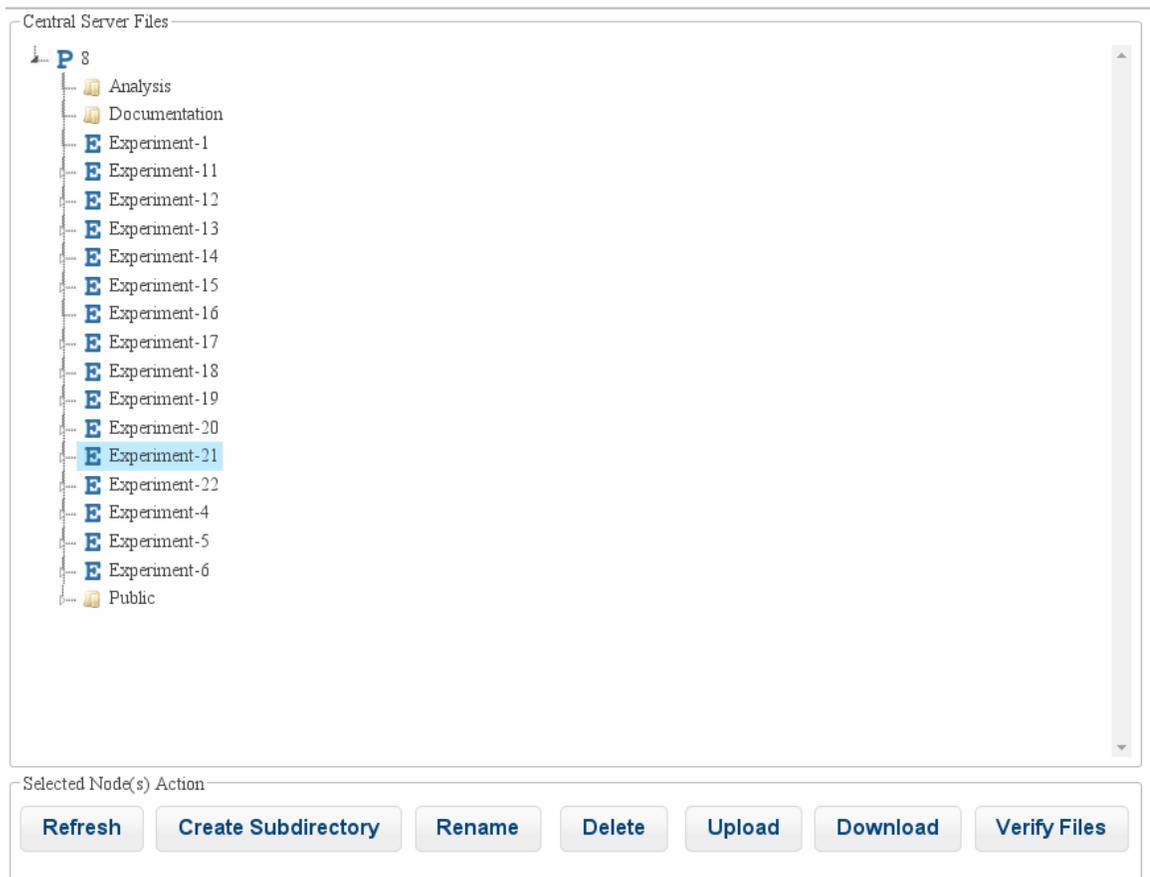


2. Click the “Stop” button and press “Send” to send a command to stop transferring data from the given client to the Central Server.

Data Upload

Prior to data upload, it is expected steps in the “Data Upload Preparation” have been completed. Also, it is expected that data has been transferred to the Central Server from clients. Here are the steps to upload data:

1. Navigate to the “Configure Upload” page. Note: It is expected that your NEES username, password, and project ID have already been entered. The project you intend to upload data for should be shown in the “Central Server Files” section.
2. Expand the directory tree under “Central Server Files” and highlight the directory you want to upload as shown below.



3. Click the “Upload” button to start uploading data from this directory on the Central Server to nees.org.