CSIR - Distributed Application Manager User Manual

101 Solutions

October 31, 2013

Version 0.7

Francois Germishuizen	11093618
Jaco Swanepoel	11016354
Henko van Koesveld	11009315

Change Log

Date	Version	Description	Done by
22 Oct	Version 0.1	Document Created	Jaco
22 Oct	Version 0.2	Added basic descriptions of all the functions	Jaco
22 Oct	Version 0.3	Added general glossary with extensions	Jaco
22 Oct	Version 0.4	Added images and minor checks	Henko
29 Oct	Version 0.5	Add detailed sys info	Francois
29 Oct	Version 0.6	Added transfer rate documentation	Henko
29 Oct	Version 0.7	Added	Henko

Contents

1 Overview

1.1 Background

The CSIR is actively developing a distributed simulation framework that ties in with various other real systems and is used to exchange information between them. The client has a number of configurations of this system depending on the requirements of the client which can involve various external applications as well.

One of the issues the client has is to quickly distribute the latest build or configuration files of their software over various computers that are needed for an experiment. In some cases the same computers may be used for other experiments which mean each of the computers may need to have various builds and configuration options.

Another issue they experience is the running, stopping and restarting of the complete simulation. During a simulation it may be determined that certain configuration options may need to be changed and distributed to the affected machines, in which case either all or some components will need to be restarted which can become tedious and time consuming.

1.2 Business opportunity

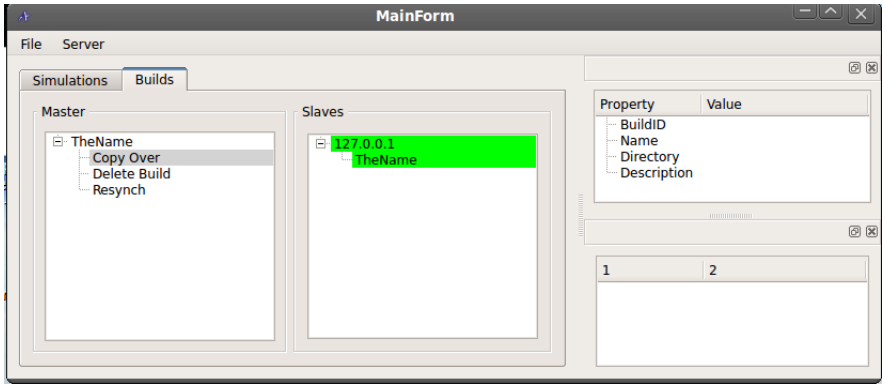
The goal of our project is to develop an application which is able to maintain various build versions of the simulation framework and distribute these builds to certain designated machines that may be required for an experiment. The application will monitor system statistics of the various machines attached to an experiment and will have the ability to execute applications on those machines which will have different configuration options.

The application will consist of a master and slave component where the master is used to control the distribution of slaves. From the master one will be able to start an experiment which will run the relevant applications on all the necessary machines.

2 Purpose of this document

This document is intended to assist users of the application, known as AppMan and AppManClient, with neccessary tasks.

The main application layout



3 Server use

3.1 Start Server

The server is started automatically, but after a manual stop can be restarted maually as follows:

- Click on the menu bar on the top of the application window
- Stop the server by clicking stop
- Start the server again by clicking start

3.2 Stop Server

The server can be stopped using this menu item:

- Click on the menu bar on the top of the application window
- Stop the server by clicking stop

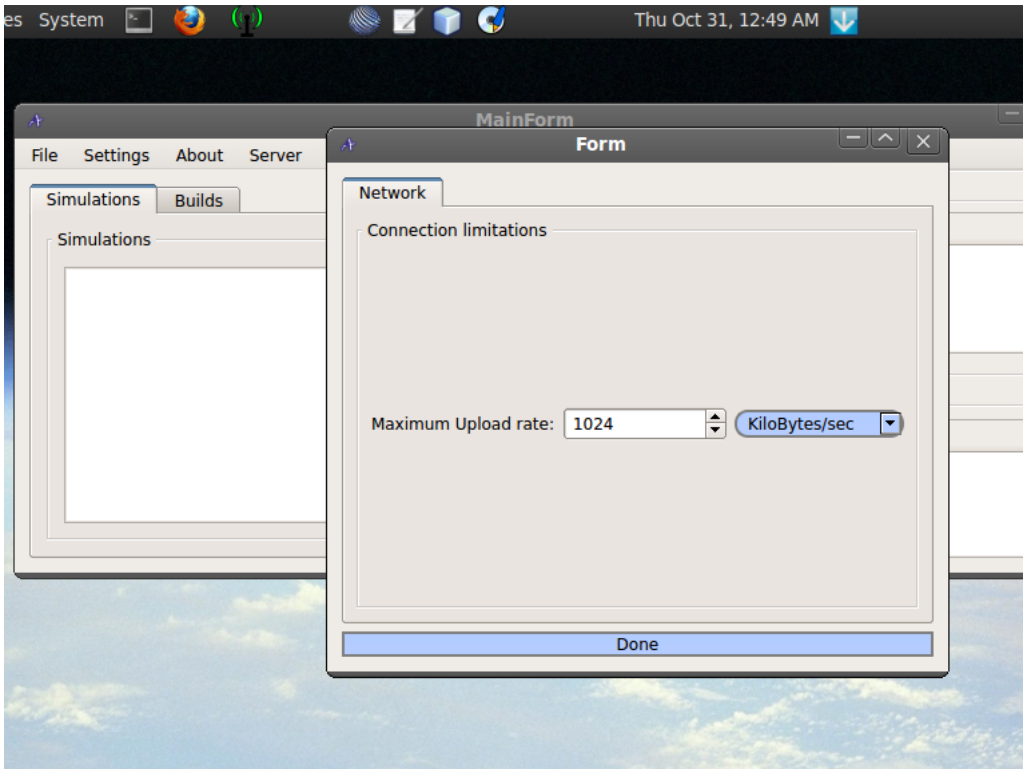
3.3 Set Port

The port can be changed from the default, effect will be taken when the server is restarted via a stop then start. The process looks as follows:

- Click on the menu bar on the top of the application window
- Click set port to set the port of the application
- Note: after that the port will be set and server will be listening on another port

3.4 Limiting transfer rate

You can limit the transfer rate by clicking "Settings"::"Application Settings" which will open the following window(see image below). This form can be used to limit the transfer rate of the application if needed.

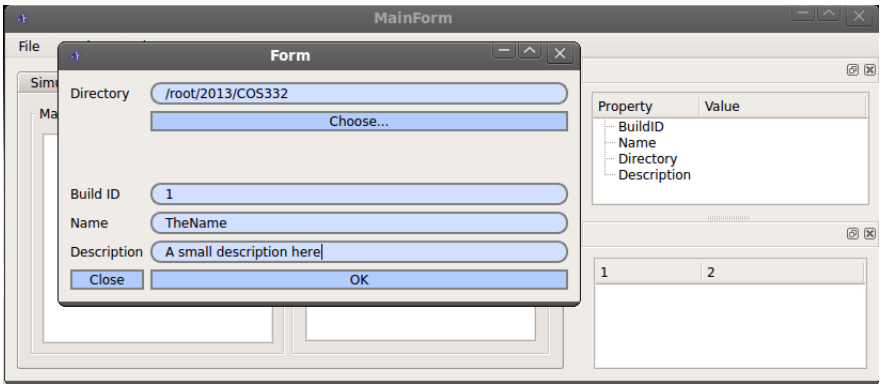


4 Build related

4.1 Add Build to AppMan (Master)

A build can be added on the master as follows, all the info, except build description is compulsory.

- Either click "File" :: "Add build" or drag and drop a folder onto the form which will open the following window and just enter the required details for the build.



4.2 Build syncrhonisation to AppManClient (Slave)

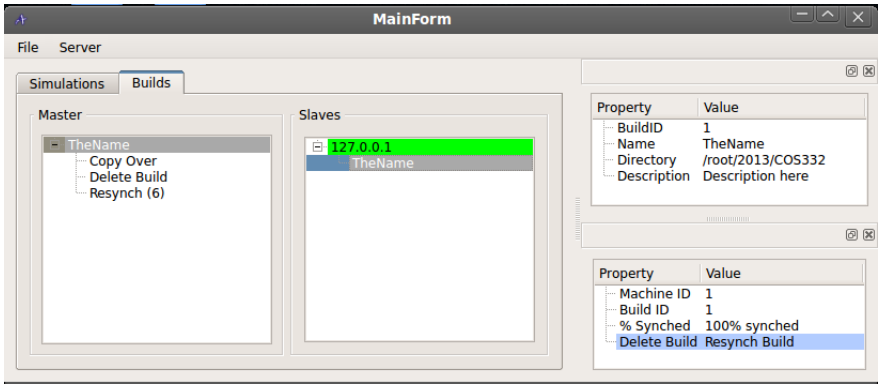
The build can be copied in the following way:

- Click a build to see the menu and click Copy over or,
- double click the build you wish to copy over and then click copy over or go to "File"::"Copy build over".
- Enter the slave ip address you want to copy the file to

After a file changed in the build the changed files will automatically resynchronise after 25 seconds, but you can invoke it directly by:

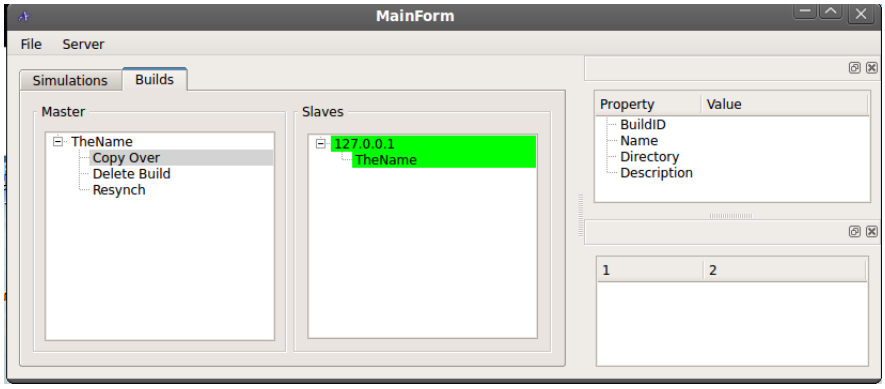
- Double click the build you wish to resynch or click on the specefic machine where you want to resynch the build

Here is the menu where the build will resynchronise after 25 seconds and there is 5 seconds remaining. You can either click the resynch button to the right or directly under the build



Here is the menu of the build on the left hand side with the following options

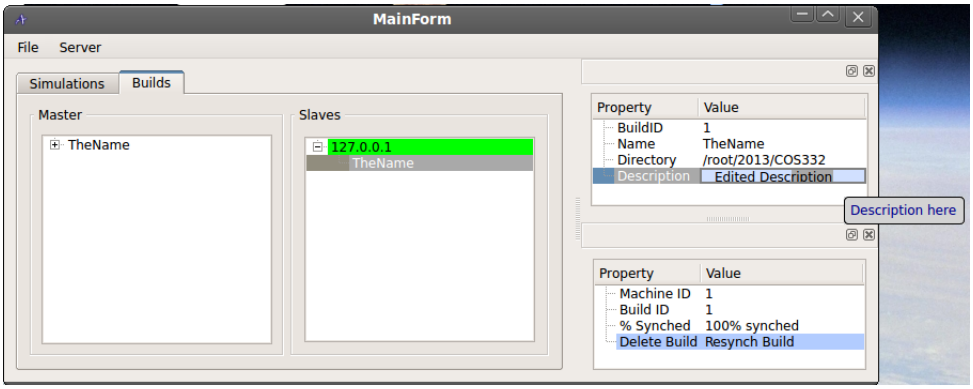
- Copy Over
- Delete Build
- Resynch



4.3 Build information editing

You can edit the information for the build you just created.

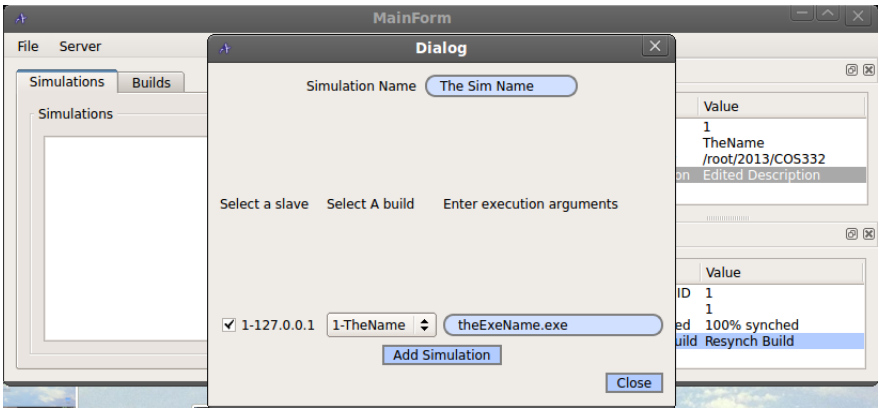
- Click the build you want to edit
- Double click the value on the right hand side as shown in the image below. The image below the description is edited. You are only able to change a small amount of things.



5 Simulation related

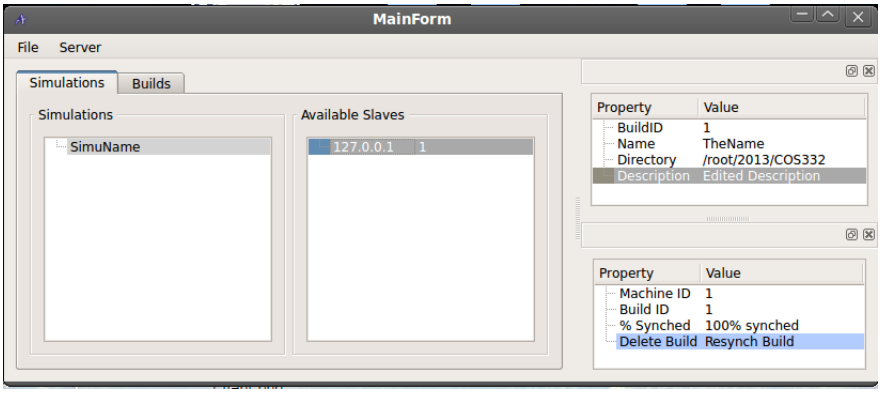
5.1 Add Simulation

To add a simulation one must have at least one slave added. The available slaves are displayed as checkbox items. A build or application can then be selected via a drop down box. If necessary command line arguments are then typed in. Once all the information is captured one can add the simulation. See below:



5.2 Execute Simulation

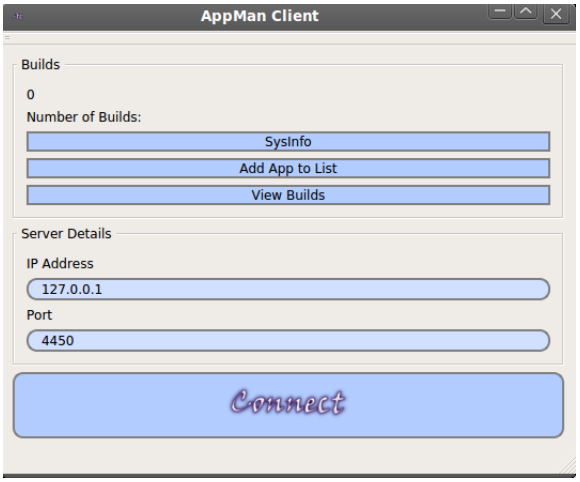
A simulation can be selected on the treewidget, then double-click the selected simulation. This will send the argument and required build or application name over the network, along with the command line arguments to the necessary slave. See below:



6 Machine related

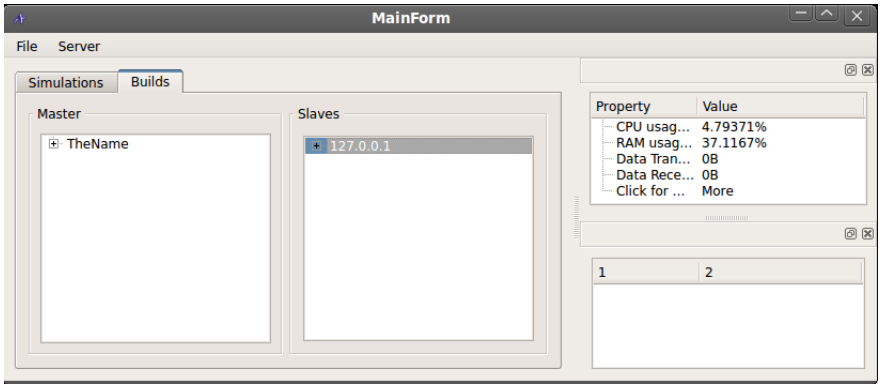
6.1 Connect

On the AppManClient one sets the port and Master IP then clicks connect. These are then stored in a settings file for future use.



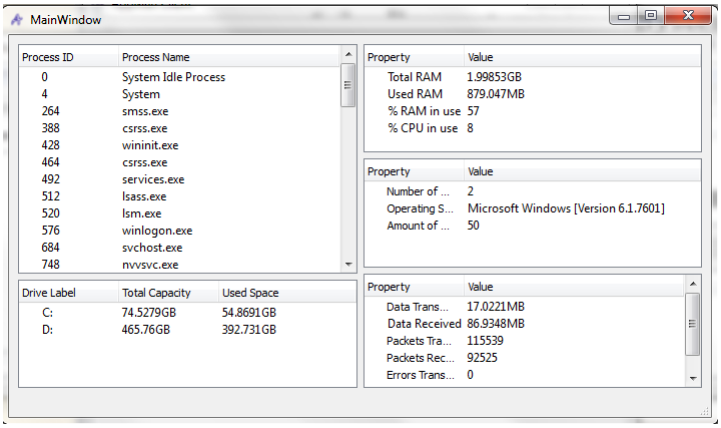
6.2 Request Minimal System Information

When a slave is clicked the request is automatically sent to the slave, then displayed in a docking widget as follows (To the right hand side you can see the slave stats displayed).



6.3 Request Complete System Information

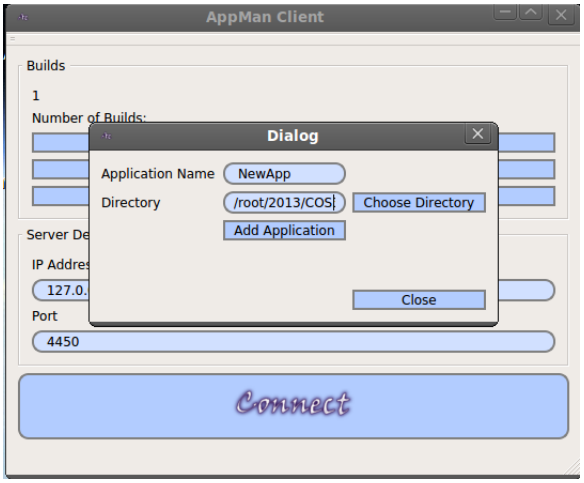
After a slave is clicked to gain minimal system information the [Click for more...] button can be pressed to reveal much more detailed system information, such as a task list and disk drive information.



7 AppManClient

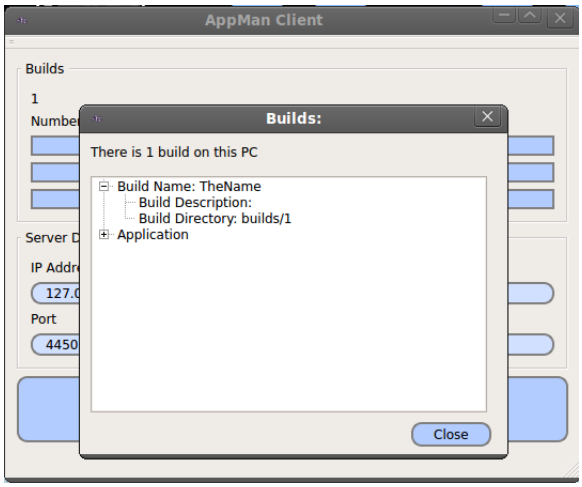
7.1 Add Application to AppList

This is done for preinstalled programs and works by clicking the add app then it will go and add the app to the list of applications



7.2 View builds

When the button is clicked, all builds and applications are displayed in a new window. See below:



8 Glossary

- Build - An application build version that could potentially be distributed to slave computers.
- Slave - A computer that will be controlled via a master computer. Application builds will be sent to this computer.
- Master - A computer that will control Slaves across a network.
- Server - A machine waiting on the network for connections from other machines.
- GUI - Graphical User Interface with which a user can control the project.
- Project - This project. The distributed application manager.
- Application - A 3rd party program the client may use during simulations