ASCOM REMOTE

Installation and Configuration

Abstract
Instructions for installing and configuring ASCOM Remote

Peter Simpson
peter@peterandjill.co.uk

Contents

1. ASCOM Remote Installation and Setup

1.1 Pre-Requisites

Please note that ASCOM Remote requires .NET Framework 4.6.2, which means that the operating system must be Windows 7 SP1 or later because .NET 4.6.2 is not available on earlier operating systems.

1.2 Installation

The installer provides options to install either the Remote Clients, the Remote Server or both. The installer will:

- **Remote Clients**: Install new 1 client device driver for each device type e.g. ASCOM.Remote1.Telescope, which will appear as ASCOM Remote Client 1 in Chooser.
- **Remote Server:** Install the remote driver server in your Start/ ASCOM Remote folder.
- Configure firewall permissions for the clients and or server.

2. Configuring the number of remote clients

After installation there will be one remote client of each device type, but you can configure the number of clients in each device type through the "Remote Client Configuration" utility that will be found in your Start / ASCOM Remote folder. This is to support complex configurations where there may be multiple devices of same type, such as cameras, focusers and filter wheels.
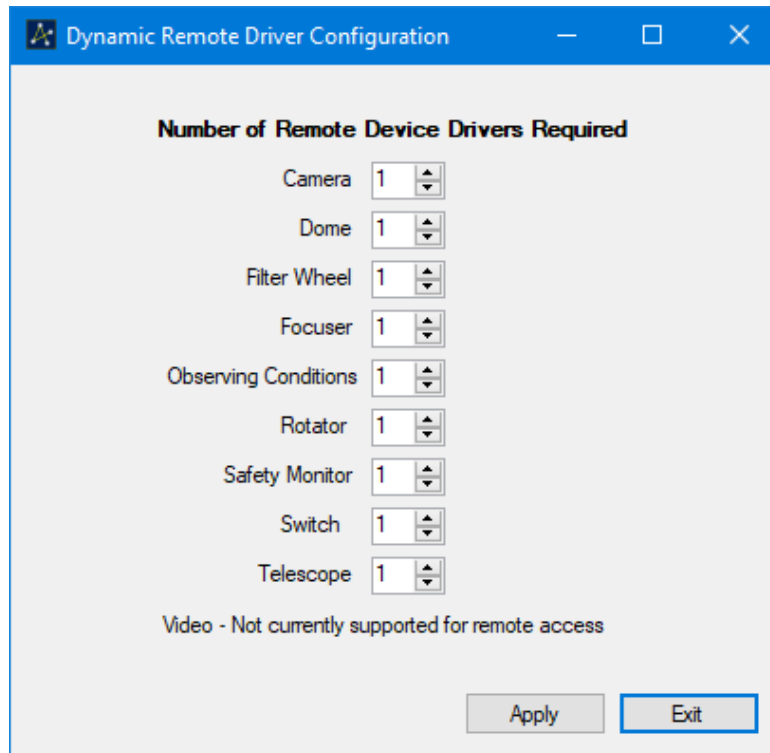

Figure 1 - Utility to configure the number of drivers of each device type

These will appear as normal ASCOM drivers named ASCOM Remote Client 1, ASCOM Remote Client 2 etc. and can be selected and configured for each application through Chooser in the normal way.

The remote clients are all hubs in their own right and can support connections from multiple clients.

3. Configuring Remote Clients

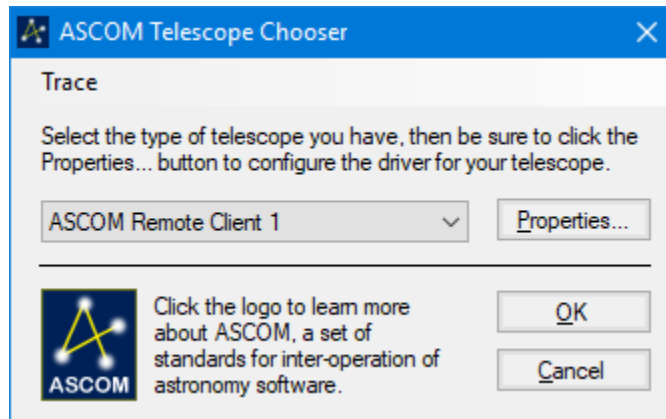The Remote Clients appear in Chooser as normal ASCOM drivers:

Figure 2 - Telescope Chooser showing a remote client

and can be configured through Chooser's Properties button in the usual way.
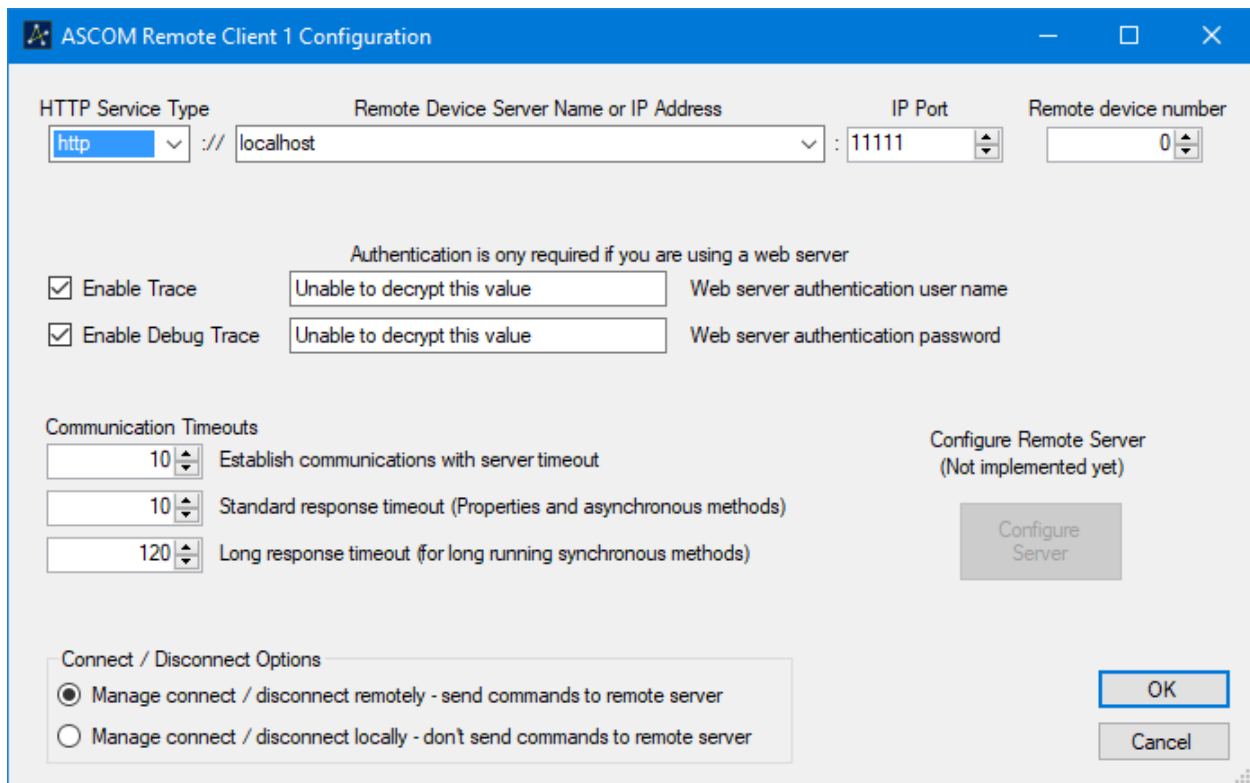

Figure 3 - Remote client configuration form

The service type (HTTP/HTTPS), IP address or host name and port set on the configuration form must match the values used when configuring the Remote Server.

There are three communication timeouts, one for establishing an initial connection with the Remote Server the second for relatively quick response commands such as CanXXX properties and a third for slow response commands such as Telescope.SlewToCoordinates. The standard response timeout

default should suit most requirements, but you may need to increase the slow response timeout depending on the longest command completion time expected under normal circumstances for your remote device.

3.1.1   Authentication

The username and password fields allow authentication credentials for the remote server to be configured. Any values entered are encrypted before being persisted in the Profile. These fields are only useful if Apache or some other web server is used to proxy incoming remote device server connections and it has been configured to require a password to access the remote server URIs.

Notes

- For testing its fine to run the driver and the remote server on the same PC and to use 127.0.0.1 as the IP address for both clients and server
- The remote server itself only supports HTTP, see Section **Error! Reference source not found.**

4. Configuring the Remote Server

Run the server, which is located in Start/ ASCOM Remote, and use the "Setup" button to start the setup dialogue enabling served device types and devices to be selected.
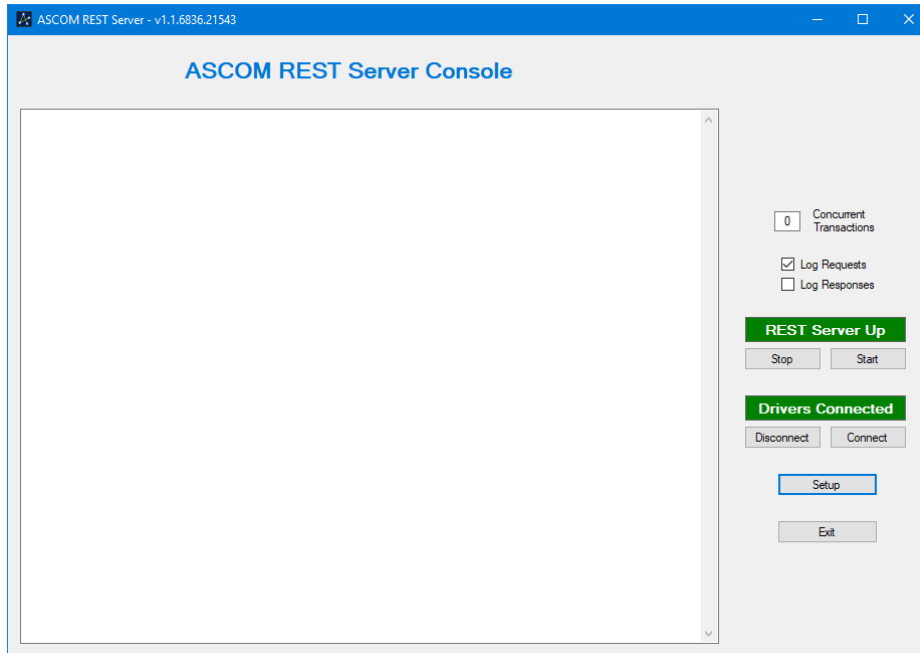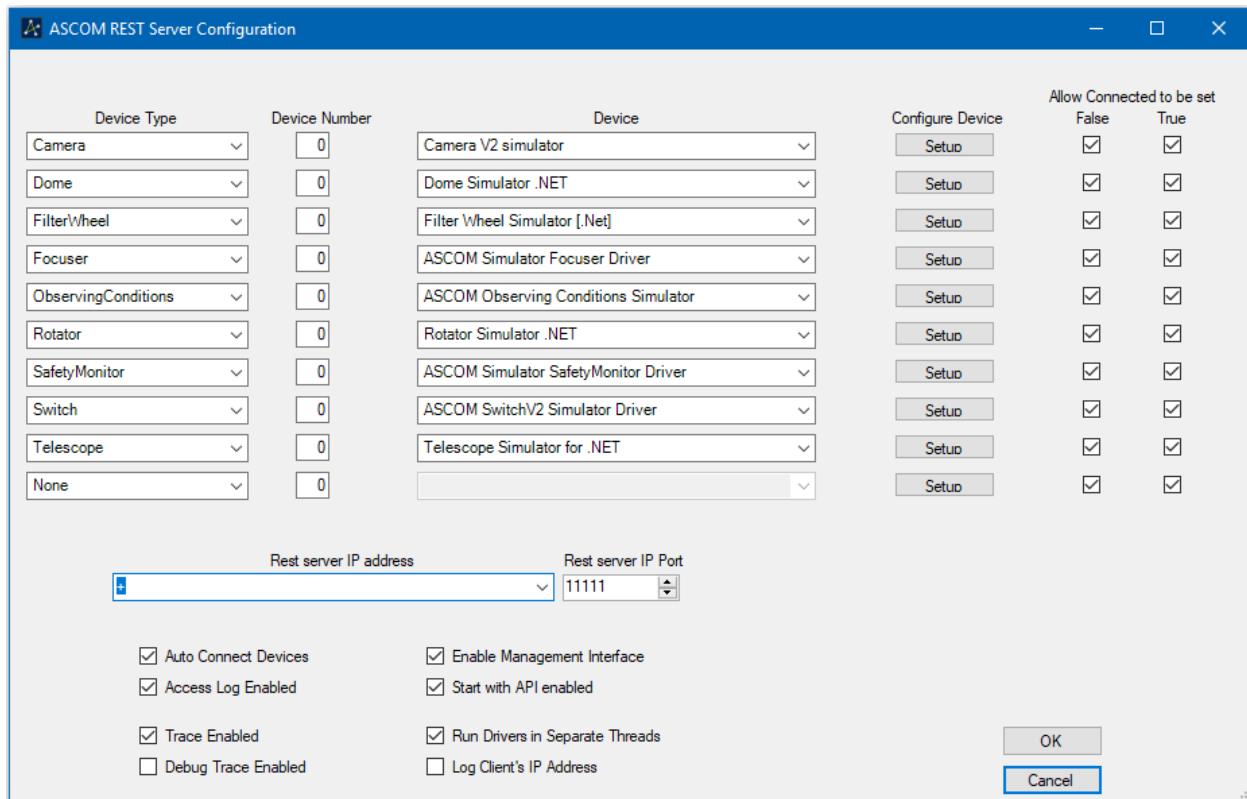

Figure 4 - Remote server console

Figure 5 - Remote server configuration form

4.1.1 IP Address

The REST Server IP Address and Rest Server Port dropdowns enable you to select the IP address and port number on which the server will listen. The IP address list will be pre-populated with all the available network addresses on the host PC plus "localhost".

If you have more than 1 network adapter on the PC you can also use the "+" or "*" addresses in the dropdown list to listen on all configured network ports.

4.1.2 Device Selection

To set up a device to be remotely served, first select the type of device in one of the "Device Type" drop-down boxes, then select its driver from the corresponding "Device" drop-down box. Make sure that all unused "Device Type" drop-downs are set to "None".

"Device Numbers" are automatically assigned as device types are selected and relate to the number of devices of that specific device type that are configured. E.g. the first focuser driver that is configured will be focuser device "0" while the second focuser device will be focuser device "1" etc.

The configured "Device Number" and "Device Type" uniquely identifies a remote device and **it is these that must be configured in the remote client** to specify the required remote device.

4.1.3 Connected Settings

The remote device server can be configured to honour, or disregard, Connection property change instructions from clients. This enables a device to be maintained in a connected state even if the client disconnects, which can be useful when there are multiple concurrent clients or when the device is intended as an "always available" service.

When the "Connected" check boxes are unset, client drivers will see Telescope.Connected changing state as they expect, but the state of the remote device will not change.

4.1.4 Management Interface

The management interface provides some capabilities to manage the server remotely:

- Return the ASCOM Profile on the remote PC
- Return the Remote Server device configuration
- Return the instantaneous number of concurrent transactions that the server is handling
- Unload and reload the served drivers

The management interface is enabled or disabled through the "Enable management interface" check box.

4.1.5 Drivers in Separate Threads

The "Run Drivers in Separate Threads" checkbox chooses between:

- Running each driver in its own separate thread with its own independent Windows event loop. (Default)
- Running all drivers on the Remote Server's main thread sharing a common Windows event loop.

Running drivers in their own threads is the preferred mode of operation because it provides greater isolation of driver issues from other drivers and from the Remote Server itself. There are currently no

known downsides to this approach; the run all on the main thread option, however, is provided as a fall back in case of issues arising when using separate threads.