

TestManager – Administration Guide

RedRat Ltd

July 2015

For TestManager Version 4.57



Contents

| | | |
|-------|--|----|
| 1. | Introduction | 3 |
| 2. | TestManager Setup Overview | 3 |
| 3. | TestManager Roles | 4 |
| 4. | Connection to the TestManager Database..... | 5 |
| 4.1 | Using SQL Server – Multi-user Mode | 5 |
| 4.2 | Using an Internal Database – Single-user Mode | 5 |
| 5. | Managing IR Signal Datasets..... | 5 |
| 5.1 | Capturing a New IR Dataset..... | 6 |
| 5.2 | Inserting the Dataset into the TestManager Database | 7 |
| 5.3 | Adding or Changing the IR Signal Names | 7 |
| 5.4 | Changing IR Signals Output via Virtual Remote Controls | 8 |
| 6. | Configuring TestManager | 8 |
| 6.1 | RedRat Hardware | 8 |
| 6.1.1 | Add..... | 9 |
| 6.1.2 | Edit..... | 9 |
| 6.1.3 | Delete | 10 |
| 6.2 | IRNetBox Output Groups (can be skipped) | 10 |
| 6.3 | STBs/TVs | 11 |
| 6.3.1 | Wiring and Testing irNetBox IR Emitters | 11 |
| 6.3.2 | Using the TestManager STB Configuration Dialog | 11 |
| 6.3.3 | Graphical Representation of the STB Layout..... | 13 |
| 7. | Backing Up the TestManager Database..... | 15 |
| 7.1 | Multi-User Database Backup..... | 15 |
| 7.2 | Single-User Data Backup | 16 |

1. Introduction

TestManager is an application to support automated and interactive control of consumer electronics equipment. Automated control takes places via scripts and to support testing of the CE equipment through the output of sequences of infrared remote control commands. Interactive control uses on-screen virtual remotes or physical handsets to route commands to selected CE devices.

The concept of using scripts to automate sequences of remote control operations is quite intuitive, but the power of the TestManager application is its ability to execute the same scripts on banks of different equipment. The key to this is the correct configuration of the test hardware and software, which is discussed in detail in this guide.

TestManager comes in two configurations; multi-user mode and single-user mode:

Multi user: Configuration and user data is stored in database that can be accessed by multiple clients allowing the creation and validation of test scripts from any computer. Typically however only one computer will be used to control the hardware under test to prevent users interfering with running tests.

Single user: Uses an internal database for all configuration and user data, so is not accessible by any other clients.

2. TestManager Setup Overview

The TestManager application (multi-user mode) uses a typical client-server architecture; the server being the database and the client application running on one or more PCs. In addition, the test setup requires one or more pieces of RedRat hardware (irNetBox or RedRat3) configured to control the equipment under test as shown in Figure 1. The elements of the setup are:

Test Farm: The set-top boxes or other consumer electronics equipment under test.

irNetBox (or RedRat3): RedRat hardware that sends infrared remote control signals to control the equipment under test.

TestManager Database: Holds all configuration and user data for the TestManager application. This is typically installed on the test room PC, but it can be installed on any suitable computer. It uses the MSDE database engine from Microsoft, which is basically a desktop version of SQL Server.

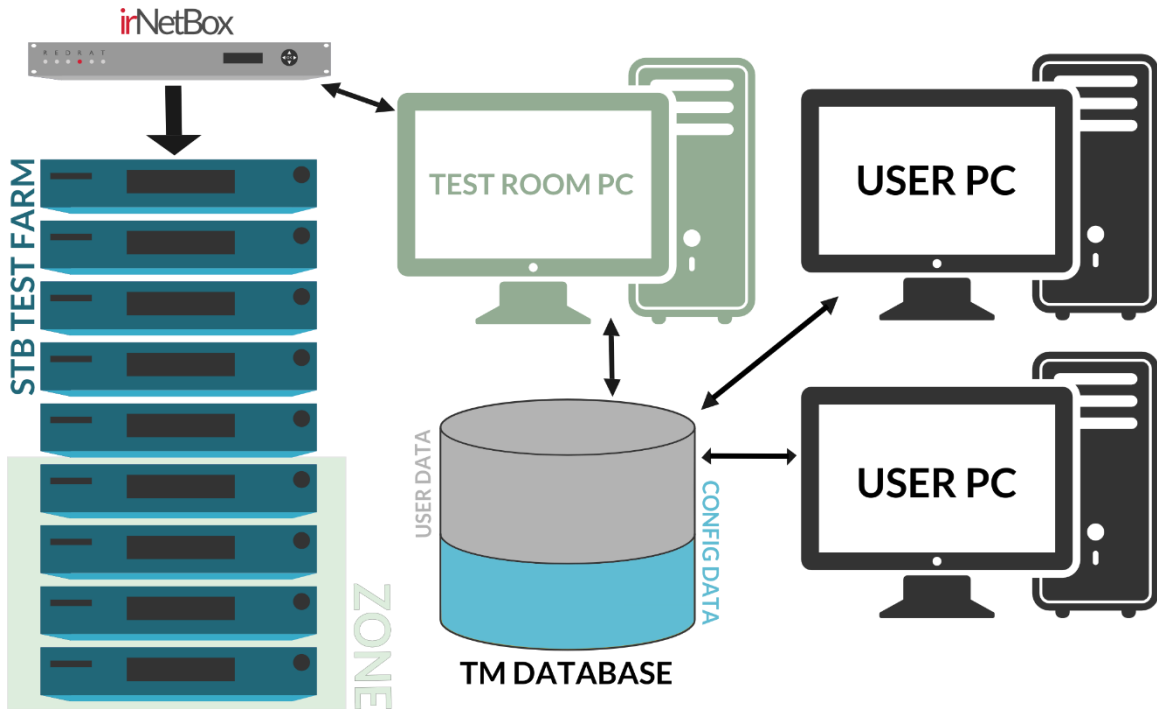


Figure 1. TestManager architecture (multi-user mode)

Test Room PC: The computer in the test room or near the equipment under test. In each installation, only one PC is able to actually execute the tests so as to prevent multiple concurrent tests being executed simultaneously.

User PC: Any other PC on the network that has the TestManager application installed and that can access the TestManager database. Test scripts can be setup and verified on user PCs, but not executed.

Zones: A subset of the full test farm for the case that tests are to be run on just certain pieces of the available equipment.

3. TestManager Roles

There are two roles when using TestManager in multi-user mode: **Administrator** and **User**. This is enforced by the database login used by a particular user, but is also dependent on how the TestManager database has been configured (see the Installation Guide).

The administrator is responsible for installing the irNetBoxes, connecting them to the STBs under test and then configuring TestManager so that it knows the mappings from irNetBox output to STB. This need only generally be done once, however the addition or removal of any

STBs will require changes to the configuration. Regular backups of the TestManager database are also the responsibility of the administrator.

Users can create and run test scripts, but do not usually have permission to change the test configuration.

In single-user mode, the user has both Administrator and User permissions.

4. Connection to the TestManager Database

When TestManager is started, it needs to load data from the data store, however the first time that the application is run it does not know where to find the data, so this has to be setup.

4.1 Using SQL Server – Multi-user Mode

In this mode an external database is used, which may be located on another server. Open the *Options Dialog* from the *Edit* menu and then enter the database login information in the *Data Source* tab.

If you have a TestManager multi-user license then you are able to use an internal database as well, however this is in effect operating in single-user mode (see the next section).

4.2 Using an Internal Database – Single-user Mode

When TestManager is first run in this mode, it will create a file for storage of your configuration data, signal datasets, scripts etc. The default name is ***TestManagerDB.sdf*** and it is initially located in your application data folder.

To view the file's location, use the *Data Source* tab in the *Options* dialog. It is possible here to:

- Open a different file: Press on the “...” button, navigate to the new file location, select and open it.
- Create a new file: Press on the “...” button, navigate to the location of the new file, enter a file name in the *File name* field and press *OK*.

It is possible to have many different files, although this is not necessarily recommended as single file can store all necessary configuration information.

5. Managing IR Signal Datasets

All IR signal data is held in the TestManager database rather than in an external file. On installation, the database does not contain and IR signal data, so this needs to be added. If you have already used RedRat hardware to capture and output IR data, then the XML datasets can simply be loaded into the database (see section 5.2).

TestManager uses a fixed set of IR signal names as keys to look up the correct IR signals when executing scripts. For standard STBs the signal names initially configured are:

| | | |
|---|--------|----------|
| 0 | Red | Program- |
| 1 | Blue | Program+ |
| 2 | Yellow | Vol- |
| 3 | Green | Vol+ |
| 4 | Up | Mute |
| 5 | Down | TV-Guide |
| 6 | Left | Text |
| 7 | Right | Menu |
| 8 | OK | Info |
| 9 | Power | Exit |

Table 1. IR signals names for Freeview STBs

These signals need to be captured from the remote of the new STB using the Signal DB Utility and then inserted into the TestManager database.

5.1 Capturing a New IR Dataset

If you have a RedRat3 available, it is easier to use this rather than an irNetBox to capture IR signals (due to better light filtering and different method of communication). Before attempting to capture the whole dataset, it is worth experimenting with one signal to ensure that the capture and replay works correctly. If it does not appear to work, then please visit the RedRat support forums or contact RedRat support.

The following steps use the *New Device/Remote Wizard* in the Signal Database Utility to capture a new dataset:

1. Select *New Device/Remote Wizard* from the *Edit* menu in the Signal Database Utility and fill in the fields in the dialog, selecting SET_TOP_BOX as device type.
2. In step 2 of the wizard, adjust the list of signals to be the same as those given in Table 1 unless the STB does not support this functionality.
3. In step 3 of the wizard, press the *Start Capture* button to initiate input of the signals from the remote.
4. Once complete, save the XML file for use by the TestManager application (and to keep as a backup).

5.2 Inserting the Dataset into the TestManager Database

IR datasets are loaded on top of any data already in the database, however no data will be deleted. Therefore if device A, signal B already exists in the database and the file you are loading also contains device A with signal B, then this will overwrite the database version. In this way it is possible to update signal data in the database if any of the signals in the database don't work effectively.

The signal data XML file can be loaded using the *File* → *Load Signal DB* menu item.

5.3 Adding or Changing the IR Signal Names

TestManager will only recognize certain IR signal names in scripts or when being output from the on-screen remote control, these names being held in the TestManager database. The reason for using a pre-defined set of signal names is twofold:

1. To enforce a standard naming convention across all STB IR signal datasets in an installation.
2. To ensure that signal names, macro names and named operations are all unique.

Table 1 shows the standard IR signal names for Freeview STBs in the UK. When TestManager is used with other types of STBs then the set of IR signal names may need to be adjusted, which can be done through the *IR Signal Names* dialog (launched from the *Edit* → *Configure* → *IR Signal Names...* menu item) as shown in Figure 2.

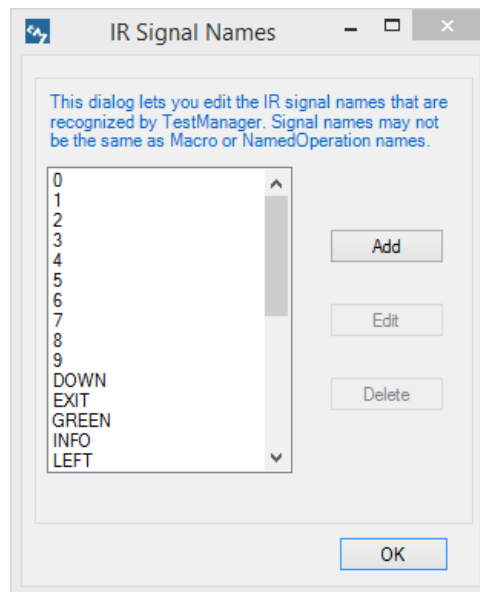


Figure 2. The IR Signal Names dialog.

For example, if you want to control a new STB that has a **Services** button on the remote control, then this name would need to be added to the IR signal name list. Once the IR dataset for this

remote control has been captured and loaded into *TestManager*, then the **Services** signal can be used in scripts, macros and from the on-screen remote control (if this has a **Services** button).

5.4 Changing IR Signals Output via Virtual Remote Controls

The keys on the virtual remote controls each output an IR signal of a given name, however in some environments the IR signal name associated with a key may not be correct. For example, STB remotes use “OK” and “Select” interchangeably, but for scripts to be run against all STBs, a choice has to be made as to which of these two is going to be used in scripts.

If the on-screen remote control buttons attempt to output IR signals with the wrong names, then the IR signal names associated with each button can be changed. When using *TestManager* in the *Admin* role, right click on any remote control button and select the *Set Button Action* menu item. This brings up a dialog in which the IR signal name that is associated with that key can be changed.

Once changed, the new key mapping is stored in the TestManager database and will then be used whenever this remote control is shown on the screen.

6. Configuring TestManager

It is assumed that the following steps have already been taken:

1. irNetBoxes have been installed and connected to the network so that they can be discovered by the TestManager application. See the irNetBox documentation for further details.
2. For multi-user mode, the TestManager database and application has been installed and setup so that the application can communicate with the database. See the TestManager installation instructions for further details.

In some of the configuration steps it will also be useful to have the IR Signal Database Utility installed, which is available on the TestManager CD or can be downloaded from the RedRat website.

The configuration options described in this section are found under the *Edit → Configure* menu item in the TestManager application.

6.1 RedRat Hardware

This step sets up which RedRat hardware is to be used by this TestManager installation as shown in figure 2.

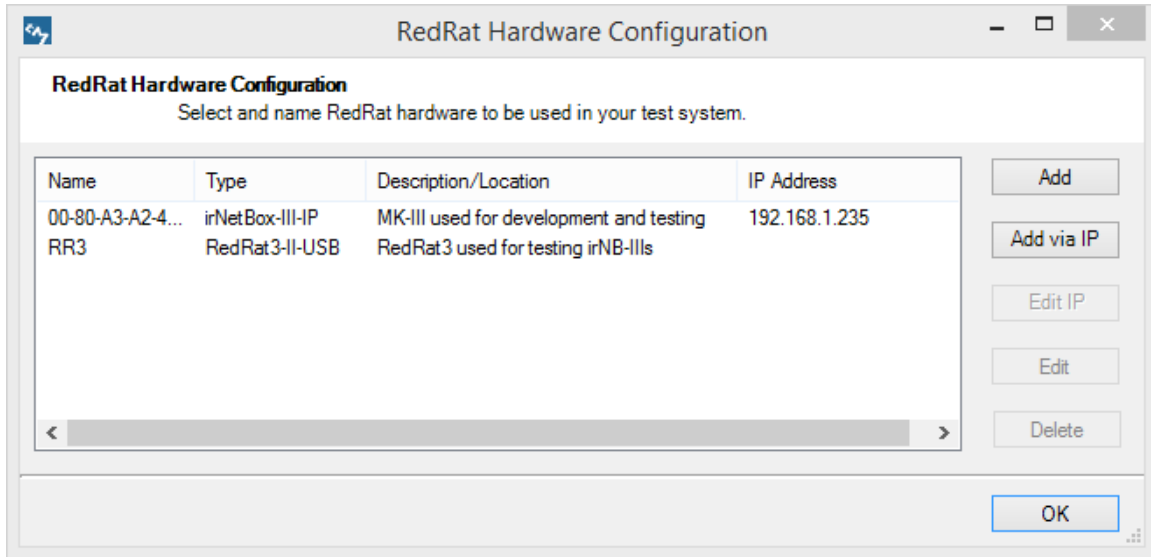


Figure 3. RedRat hardware configuration dialog.

6.1.1 Add

This initiates a search for RedRat hardware, and will discover any RedRat3s attached to this PC or any irNetBoxes on the network. Select only the hardware that is to be used with this TestManager instance. For example, if there are two test farms both using irNetBoxes, then all irNetBoxes may appear in this list, however you should only select the ones in the test farm for which this TestManager installation is being used.

6.1.2 Edit

A dialog box showing hardware specific information is displayed, such as the firmware and hardware versions. A few fields can be edited so that more useful information is displayed:

Name: Some simple name to identify the hardware. This is then used in later configuration steps.

Description: Helps identify the hardware, for example giving its exact location.

Resolve (irNetBox only): When an irNetBox is to be used, it can either be found using a UDP based network search (which takes several seconds) or by communicating directly with the stored IP address. If the irNetBox has a static IP address, then either option can be used, however if it obtains its IP address dynamically (using DHCP) then this field has to be set to *lookup* as the IP address is not guaranteed.

6.1.3 Delete

RedRat hardware no longer in use by this TestManager installation can be removed from the system. This is only allowed if no set-top boxes are currently configured to be controlled by this hardware.

6.2 IRNetBox Output Groups (can be skipped)

An output group is a set of irNetBox outputs that should operate as a single virtual IR output. This is rarely used, but is available for example in situations where you know that you will always want to send the same IR signals to a set of outputs. (Within TestManager, using Zones is a more flexible way of grouping STBs.)

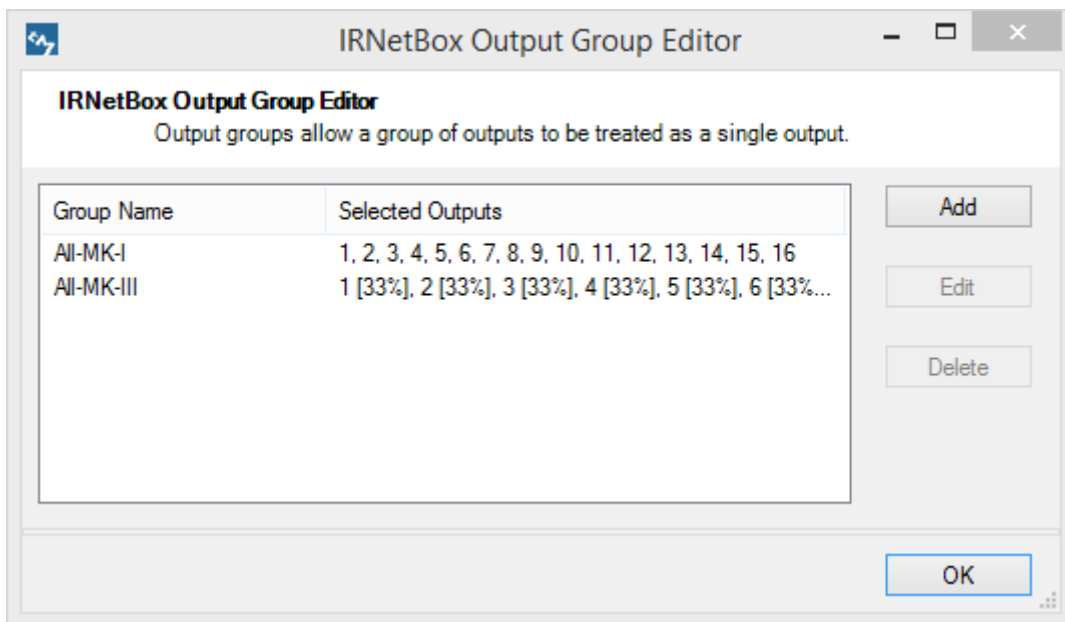


Figure 4. The output group editor.

When selecting outputs, the editor allows selection of different output powers for the different generations of irNetBox:

MK-I: Simple ON or OFF for each output.

MK-II: Power levels of LOW, MEDIUM or HIGH can be chosen by right-clicking on each output.

MK-III: Power levels of 0 (off) to 100 can be chosen, by right-clicking on an output and using the slider.

6.3 STBs/TVs

In this step each STB/TV is assigned a particular IR output and an IR dataset (see section 5), which is usually done while wiring up the irNetBox IR emitters.

6.3.1 Wiring and Testing irNetBox IR Emitters

When using an irNetBox, an IR emitter (also sometimes called “flasher”) is stuck to the front of the STB as near as possible to the IR detector. It is very important that the emitter is placed correctly to ensure reliable control of the STB. On some STBs, the exact position of the IR detector can be difficult to find, so the following procedure can help:

1. Try and identify the position of the IR detector on the STB using any visual cues (IR windows etc.). Using an LED or halogen torch can sometimes help view behind any IR transparent windows on the front of the STB.
2. Ensure that you have the IR signal dataset captured for this STB. See section 5 for details on IR dataset management.
3. Start the Signal DB Utility and load the dataset for use with this STB (or all datasets if they are stored in one file).
4. Plug in an IR emitter into a free output on the irNetBox that is going to control this STB.
5. In the Signal DB Utility, select the irNetBox from step 4 from the *RedRats* menu item. Open the irNetBox output selection window from the *View → IRNetBox Output Config.* menu item and then enable the output into which the IR emitter was plugged in step 4 (click on the virtual LEDs to turn on/off).
6. To help the correct positioning of the IR emitter on the front of the STB, the Signal DB Utility can repeatedly output IR signals while the IR emitter is moved around to find the optimal location. Select a suitable signal that will cause an action each time it is output (e.g. Program+), right-click on it and select *Test Signal Output*. In the output dialog select *Repeat Start*.
7. Once the optimal position has been found, stick the emitter in this position.

If it is difficult to find any position that works very reliably, then try using output 1 on the irNetBox with high-power output enabled.

6.3.2 Using the TestManager STB Configuration Dialog

The main editor is shown in Figure 5, in which STBs can be added, edited and deleted. For each STB to be controlled by TestManager, the dialog box shown in Figure 6 is used enter the following information:

Name: Some simple name to identify the STB. This is then used to identify this STB in zones and scripts.

Description: Helps identify the STB, for example giving its exact location.

AVDeviceName: This is the name of the IR signal dataset that will be used with this STB. A drop down list allows the choice of any IR datasets stored in the TestManger database – see section 5 for details of loading IR datasets.

IROutput - RedRatID: Selection of one of the RedRat3s or irNetBoxes configured in section 6.1.

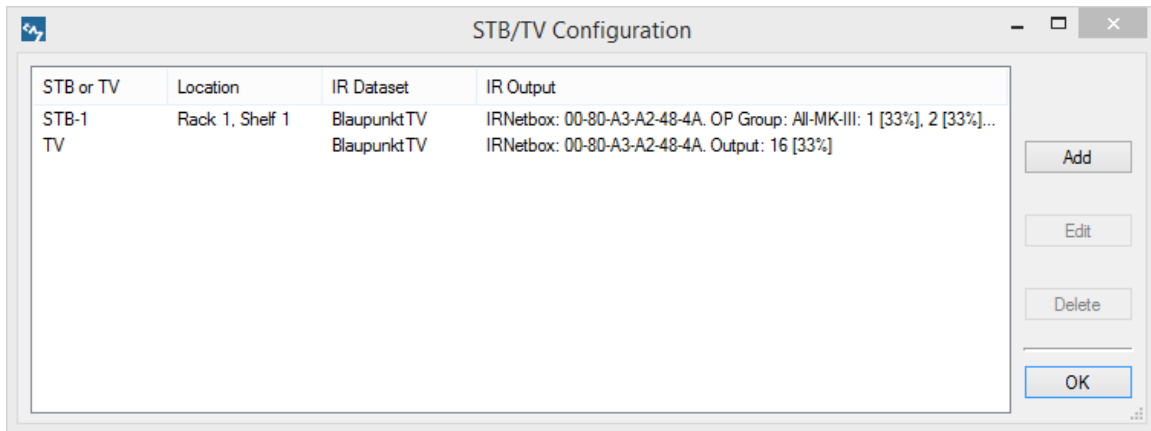


Figure 5. The editor for setting up STBs

IROutput - OutputNo: If the RedRat is an irNetBox, then this lists the outputs available. Note: Output 1 has two power modes, normal or high-power.

IROutput - OutputGroup: If an output group has been configured as in section 6.2, then it can be selected here for use with this STB.

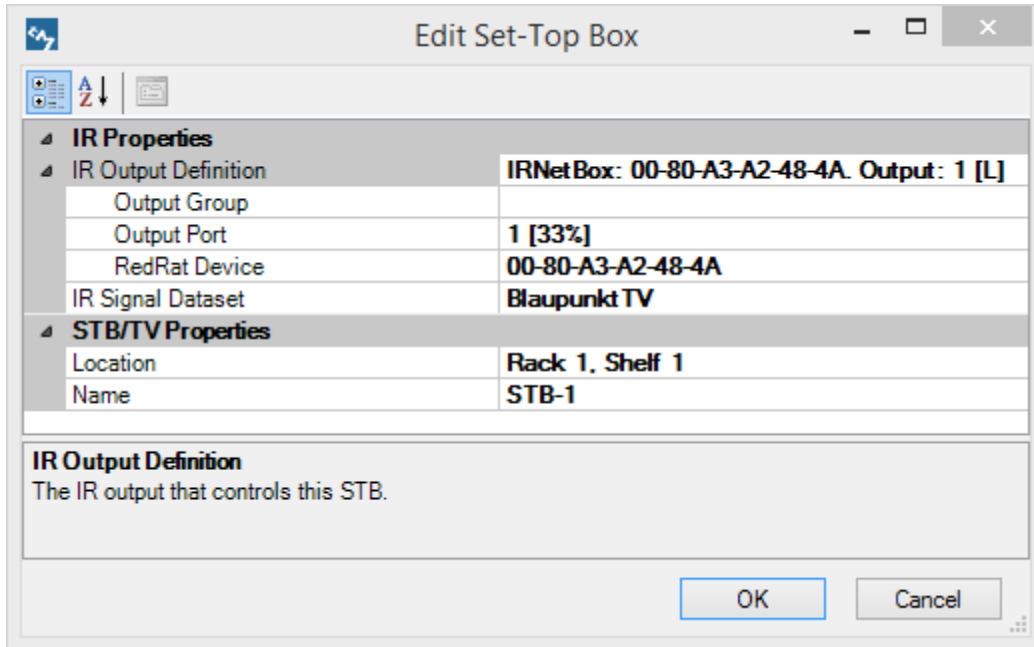


Figure 6. Configuration dialog for one STB.

6.3.3 Graphical Representation of the STB Layout

In some contexts, having an intuitive and rapid method of interactive STB control for all STBs in the system can be useful. The *STB Layout* window is designed for this purpose and clicking on any STB in the layout window will bring up an on-screen remote control to directly control that STB. However it is not required for the operation of scripts, so this section can be skipped and returned to at a later point in time if required.

Figure 7 shows an example of a *STB Layout* dialog, which has STB icons arranged to reflect their groupings and/or physical positions. Once configured, clicking on any STB icon will bring up an on-screen remote control to that STB, so allowing simple control of any STB within the test system. This section explains how the *STB Layout* window is configured.

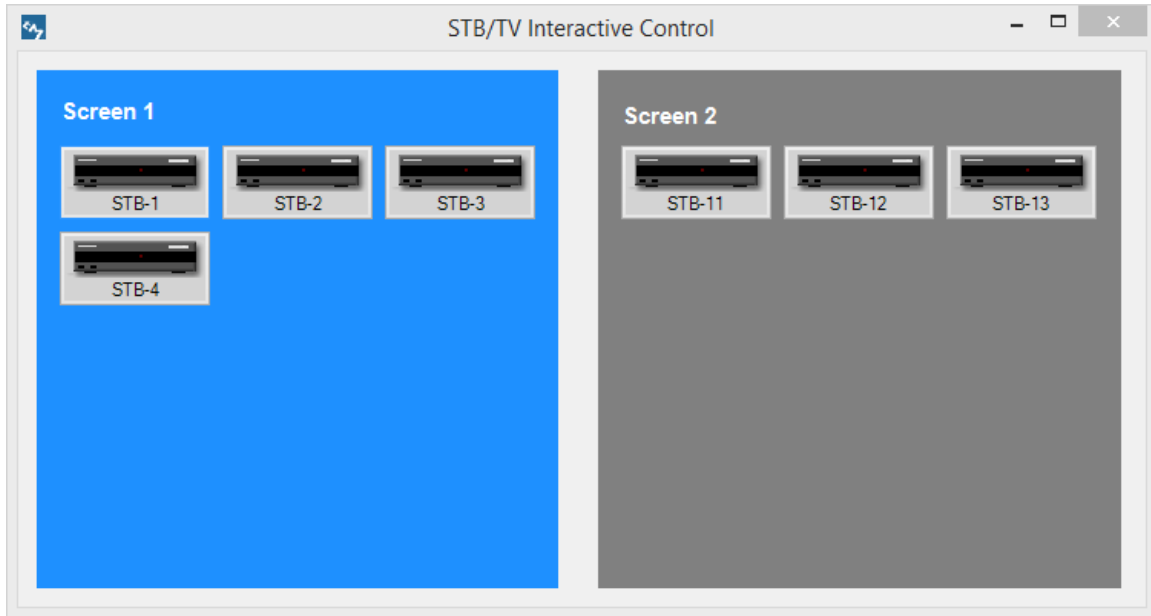


Figure 7. The Graphical Representation of STB Configuration

If you are using *TestManager* in *Administrator* mode, then the STB Layout window can be switched into edit/designer mode, where STB icons can be added and configured. To start the designer, do one of the following:

- Open the STB Layout window from the *View* → *STB Layout* menu item. Right-click in the window and select *Edit STB Configuration* from the popup menu. Or...
- Use the *Edit* → *Configure* → *STB Layout* menu item.

6.3.3.1 Adding Panels and Labels

Panels and labels can be added by right-clicking on the editing panel.

6.3.3.2 Adding STB Icons to the Diagram

The STB Layout window works with STBs that have already been defined, as described in section 6.3.2. To add an STB to the diagram, right-click and select *Add* from the popup menu, and an STB icon will be added with the name of the first STB in list of STBs known about by *TestManager*. The actual STB associated with this icon and other properties can be configured by selecting the icon and modifying the values in the grid on the right hand side of the designer.

6.3.3.3 STB Icon Properties

When an STB is selected, some of its properties can be changed directly in the designer, and some in the grid on right hand side of the designer.

- **Size & Position:** Select and drag the icon to the correct position. Also, the corners and sides can be moved to adjust its size.

- **STB/TV Name:** The actual STB that this icon represents. The drop-down list is populated from those configured in STB/TV Configuration window as given in section 6.3.2. There can only be one icon per STB, so once an icon has been assigned to a certain STB, that STB name is removed from this drop-down list.
- **Remote Control:** Select an on-screen remote control to use to control this STB. When this STB icon is clicked by a user, this on-screen remote control will be shown.
- **Background Colour:** A background colour for icon can be selected to help differentiate it or visually reinforce groupings.
- **STB Image:** A drop-down list of available icon images allows you to select a particular image for this STB. It is also possible to select your own image for an STB by clicking on the “+” symbol in the image selection dialog.

When editing is complete, changes are stored and if the STB Layout window is open, it will be updated to reflect the changes made.

7. Backing Up the TestManager Database

The TestManager database can contain a lot of data, scripts and other information that would take a lot of time to re-create, so it is good practice to take backups of the database to store elsewhere; on a different computer at least, but ideally off-site.

There may already be backup mechanisms in place for databases and file systems, but if not then it is possible to make database backups from TestManager if logged on as a TestManager administrator. The backup process depends on whether the application is setup to run in single-user or multi-user mode.

7.1 Multi-User Database Backup

This makes a copy of the SQL Server data from the database by instructing the SQL Server itself to execute a backup operation. Figure 8 shows the dialog for creating the backup, which is started from the *File -> Backup TestManager DB* menu item. This will cause SQL Server to create the named backup file in the given directory on the database server.

Please Note: This all takes place on the database server and not on the local machine (unless they are the same).

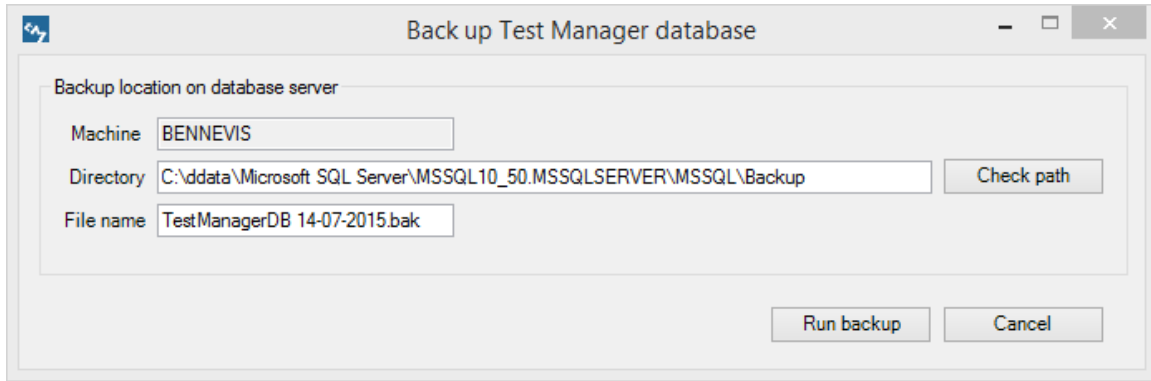


Figure 8. The database backup dialog

As this takes place on the server, to help in the selection of the directory the **Check Path** button can be used to help ensure that the directory actually exists.

7.2 Single-User Data Backup

As data is stored in a file on a local machine, the backup in this mode takes a copy of this file and prompts you for a location to save the file. It is recommended that the backup files are copied off the machine in case of hard-disk failure.