# Table of Contents

1 Introduction........................................................................................................................... 4
   1.1 Steps to simplified audio distribution........................................................................... 5
   1.2 How does it work?........................................................................................................... 5
   1.3 In more detail.................................................................................................................. 6
      1.3.1 Hardware.................................................................................................................. 6
      1.3.2 Set up a User Project............................................................................................... 6
      1.3.3 Device startup and configuration update............................................................... 7
      1.3.4 Device operation...................................................................................................... 7

2 Reflector Client Application.................................................................................................. 8
   2.1 Load firmware and activate the Reflector application. ................................................. 8
   2.2 Startup........................................................................................................................... 9
   2.3 Reflector Client Web UI.................................................................................................. 9
      2.3.1 Local IO status......................................................................................................... 10
      2.3.2 Stream details.......................................................................................................... 10
   2.4 Advanced users: configuring the client device............................................................. 10
   2.5 Encoding in Mono......................................................................................................... 11
   2.6 Device LEDs / Display................................................................................................... 12
      2.6.1 Status LEDs / Display............................................................................................ 12
      2.6.2 I/O LEDs................................................................................................................ 12
   2.7 Auto Failover playback.................................................................................................. 12
   2.8 Flag streaming error on local relay............................................................................... 13

3 The Reflector service web UI.............................................................................................. 14
   3.1 Requesting an account................................................................................................. 14
   3.2 Create a project.............................................................................................................. 16
   3.3 Home page................................................................................................................... 20
   3.4 Viewing project status.................................................................................................. 21
   3.5 Modify device settings.................................................................................................. 22
   3.6 Modifying Account Details......................................................................................... 23
   3.7 Menu options................................................................................................................. 23
      3.7.1 Home...................................................................................................................... 23
      3.7.2 Projects................................................................................................................... 23
      3.7.3 Account Details...................................................................................................... 24
      3.7.4 Search.................................................................................................................... 24

4 Legal Information.................................................................................................................. 25
1 Introduction

The Reflector service is a way for Broadcasters to quickly and easily set up a unidirectional STL, Remote Contribution station or Syndication network using the internet, but without all the complexities of configuring IP networks, or punching holes through firewalls: no specialist IT knowledge is required to set up and use the service.

Barix devices being used with the Reflector service must have the latest Barix Reflector Client application loaded and active. The application is contained within both the ABCL_Reflect_xxx and ABCL_STL_xxx firmware packages, which can be downloaded from the Barix website.

The Barix Exstreamer 500 and 1000 series devices come preloaded with both the Reflector and the more complex STL application, the user simply selects which one they wish to use. Other Barix hardware can also have the Reflector Client loaded on to them to enable use with the Reflector service.
1.1 Steps to simplified audio distribution

1.2 How does it work?

The Reflector Service is a subscription system, subject to a usage charge. A User must have an active User ID in order to be able to use Reflector. These can be obtained via the 'Request an Account' functionality on the Reflector Service homepage.

The Reflector Service only operates with devices that have been defined to it. A User must therefore set up a Project containing the details of all the Devices. Each project needs a Sending Device (Encoder) at the Head end and Receiving Device(s) - Decoders - at the transmitter(s) / affiliates. Requests from unregistered devices are ignored by the server.

The main components of the Reflector are shown in the following diagram:

- Database – stores all data about accounts, projects and devices
- Configuration server – assigns project devices to individual Reflector Servers and communicates configuration details to devices
- Reflector server(s) – receive details of devices from the config server and handle communications with them; receiving the network audio stream from the Encoder and forwarding it to the Decoder(s)
• Web server – handles the User Interface, allowing users to set up and maintain their projects, updating the database accordingly.

• Encoder/Decoders. On startup, each Reflector Client device will contact the Reflector Services defined to it in its startup configuration (US / EU ...), requesting the details of its project configuration, it then contacts the appropriate Configuration Server, applies its configuration and operates accordingly.

1.3 In more detail

1.3.1 Hardware

The sending device is a Barix device capable of encoding the input audio and sending it out onto the network: An Exstreamer 500 / 1000, or an Instreamer 100. The receiving device is a Barix device capable of decoding the network stream and outputting it as analogue/digital audio: An Exstreamer 100 series¹, 500 or 1000.

Always check that you have the latest version of the Reflector application; it can be downloaded from www.barix.com/downloads. Chapter 2 explains the client device firmware.

1.3.2 Set up a User Project

The Reflector is a web based service that requires an account to run it. Users must first obtain an account from the Service administrator via the 'Request an Account' function on the Service homepage.

After setting up a User account and having it activated (a confirmation email is sent for

¹ Exstreamer 100 /110/120: these output Analogue audio
both events), users create a 'Project' – this is a way of telling the service to associate devices with each other. Devices must then be registered to the Project, giving the unique MAC address\(^2\) of each device and a few straightforward configuration details. Chapter 3 details how to set up an account and projects in order to be able to use the Reflector service.

### 1.3.3 Device startup and configuration update

On startup, a device must first find out what its configuration is. To do this, it works its way through the list of known Configuration Servers defined to the device, pinging each in turn until it receives a positive response. If the device is unknown to a Configuration Server, the server returns an 'unrecognised device' message and the device moves to the next entry in the list.

If no positive response is received from any of the known configuration servers, the device will pause and then start querying the list again, repeating this behaviour until it gets a positive response. The device will also start failover playback of any file(s) in attached USB stick. In cases where the local IP network infrastructure is not working when the device starts up, this playback will be delayed whilst the device attempts to determine its IP address. There are various strategies for addressing this delay e.g. assigning a fixed IP address, or disabling one or more of the IP search techniques.

By default, the devices use DNS names to address the servers, so it is important either that the network they are connected to supports DHCP / DNS, or that the device is configured with the relevant network settings to be able to reach a DNS server.

When a registered device contacts a configuration server, the server responds by sending the device’s current configuration, the device receives this and applies it.

During operation, devices regularly ping their configuration server with a 'keepalive' message: this information is used to update the device status of the project on the Reflector web UI.

If a user updates the configuration of a device, the Configuration Server will immediately send an update to the device concerned, which will apply it.

### 1.3.4 Device operation

Once a device has started up and successfully received its configuration, it then operates according to the received configuration.

The principle configuration element is whether the device is an Encoder, or a Decoder (this cannot be auto-detected as several Barix hardwares can operate as both).

---

\(^2\) written on the underside label of every Barix audio device
2 Reflector Client Application

2.1 Load firmware and activate the Reflector application.

To use the Reflector Service, all devices must have one of “ABCL_Reflector_client_vxxx” or “ABCL_RTP_STL_fdx_vxxx” packages loaded: they are dual application firmwares containing both the Reflector Client and the more complex Barix STL application.

The Exstreamer 500 comes preloaded with the Reflector package, the Exstreamer 1000 with the STL. Any Barix audio hardware can run the Reflector Client application, subject only to limitations of the hardware (Instreamers cannot be decoders, for example). This guide is for the Reflector firmware, there is a separate manual for the STL Application available from [www.barix.com/downloads](http://www.barix.com/downloads) where the firmware package can also be found. The firmware is delivered as a zip file which contains a “_readme1st.txt” file that explains how to load the firmware into the hardware and there is also a ‘howto’ video on the “BarixAG” YouTube® channel.

All devices to be used must have the firmware loaded and be active in 'Reflector' mode.

When loading the firmware, take a note of the MAC addresses, you will need them later:

- it's on the top line of the UI, to the right in the picture below
- it's physically written on the underside of the device

Once loaded, to switch between the two applications, simply navigate to the 'Reboot' page of the Client web UI and select to 'reboot as' the application you want:
2.2 **Startup**

When the device first starts up, it searches for a Reflector service, whilst it is doing this, the UI shows:

![Image 1](image1.png)

if the device does not get a positive response from any of the Reflector services, it will display:

![Image 2](image2.png)

The firmware will continue to cycle through the available servers until it gets a positive response.

2.3 **Reflector Client Web UI**

When a device receives a configuration response from a configuration server, the UI will display the details and report on the device status:

![Image 3](image3.png)
The received configuration is shown in light grey on the left side of the screen, along with the timestamp of the configuration.

2.3.1 Local IO status
The Reflector Client automatically implements an IO tunnel from encoder to receiver(s) and the UI shows the state of any IO the device has.

On suitable hardware, relay 1 can be used to signal outgoing/incoming stream loss. On the web UI 'green is good', it means that the stream is OK. If the stream is missing, the relay is 'switched off' / shown grey on the UI. It is configured this way so that if power is lost, the logic for any equipment connected to this relay is consistent.

2.3.2 Stream details
In decoder mode the middle section of the UI beneath the relay status shows some details about the received stream: bitrate, detected latency and some indicators as to the quality of the link: significant numbers of lost, duplicated or dropped frames should be investigated.

2.4 Advanced users: configuring the client device
The Reflector Client should not need any configuration in most cases, however there can be some occasions when it is necessary for infrastructure reasons to configure certain settings.
Clicking on the 'Configuration' tab on the UI navigates to a screen that allows the user to set the network details for the device, should this prove necessary. If the environment supports DHCP and DNS, then typically no extra configuration is required.
Clicking on the 'Reflector' tab on the left allows the Configuration server list to be updated e.g. If a user has set up their own instance of the Reflector service. The servers are contacted in order, so for the quickest startup, set server 1 to the Reflector Service that you are using.

Clicking on 'Security' allows the user to set some basic security features such as a password for the web UI, to prevent unauthorised or accidental changes of the settings.

### 2.5 Encoding in Mono

Note that when using the Reflector for Mono transmissions, the audio input should be presented on the left channel; in mono mode the right channel input is ignored. At the receiving end, the mono audio will be presented on both output channels.
2.6 Device LEDs / Display

2.6.1 Status LEDs / Display

The device's status LEDs are used to reflect the status at a given time. Control passes between the underlying operating system at boot up and the Reflector application once the device has started and is operating normally.

<table>
<thead>
<tr>
<th>LEDs</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>☺☺</td>
<td>Normal operation: device has configuration and is sending/receiving stream</td>
</tr>
<tr>
<td>ℹ️</td>
<td>Lost contact to config server, but has configuration and is streaming</td>
</tr>
<tr>
<td>ℹ️☺</td>
<td>Occurs when the device is not streaming, either when: &lt;br&gt;- Lost contact to Reflector server, has contact with Config server. NOT streaming &lt;br&gt;- Lost contact to config and Reflector servers, NOT streaming</td>
</tr>
</tbody>
</table>

On devices with a display, text is displayed instead:

<table>
<thead>
<tr>
<th>Display</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>L1: “Stream: OK” &lt;br&gt;L2: “XXX.XXX.XXX.XXX” ... (IP address)</td>
<td>Normal operation: device has configuration and is sending/receiving stream</td>
</tr>
<tr>
<td>L1: “Stream: Warning” &lt;br&gt;L2: “XXX.XXX.XXX.XXX” ... (IP address)</td>
<td>Lost contact to config server, but has configuration and is streaming</td>
</tr>
<tr>
<td>L1: “Stream: ERROR” &lt;br&gt;L2: “XXX.XXX.XXX.XXX” ... (IP address)</td>
<td>When the device is not streaming, either because: &lt;br&gt;- Lost contact to Reflector server, has contact with Config server. NOT streaming &lt;br&gt;- Lost contact to config and Reflector servers, NOT streaming</td>
</tr>
</tbody>
</table>

2.6.2 I/O LEDs

When configured as an encoder, the IO LEDs on the front of the device show the state of the device’s inputs.

When device is configured as a decoder, the IO LEDs show the state of the local device’s relays, which are controlled by the remote encoder's inputs, with the following exception:

**Note:** in Decoder mode Relay 1 can be set either to mirror the Encoder input 1, OR to flag stream loss.

2.7 Auto Failover playback

In Decoder mode, if the incoming stream fails the Reflector client application will:

- set the device status to reflect the cause; either 'No incoming stream', or 'No Reflector service available'
- if a USB stick / microSD card is connected then the device will automatically search

---

3 | IO LEDs are only on certain hardware: Exstreamer 500/1000 & Annuncicom 1000
for a playlist file (playlist.m3u) in the root of the attached storage

- The playlist.m3u file should contain a list of MP3 files, which should also be on the storage media: these files will be played when the decoder has no network audio stream.
- As soon as the network audio stream is available again, the device playback will switch over to it, exiting failover mode.

Note: The Reflector Client will also go into failover mode on startup, if it has tried and failed to contact the Reflector servers.

2.8 Flag streaming error on local relay

Both Encoders and Decoders can be configured to indicate outgoing/incoming stream failure on a local relay: of course the hardware device must have the relay!

Owing to the nature of the service, a Decoder can flag this stream failure virtually instantaneously, but the Encoder requires up to 90 seconds to report the failure.
3 The Reflector service web UI

3.1 Requesting an account

The Reflector web UI is used to set up, control and monitor User projects. Reflector is a subscription service operated by StreamGuys, to obtain an account, the user must first subscribe, either by mail reflector@streamguys.com, or via the home page of the appropriate service:

http://us-reflector.barix.com or http://eu-reflector.barix.com

the home page is a logon screen:

Click on 'Request an Account' and fill in the details as shown on the screen below before clicking on 'Submit request'.

Filling in the form creates a request that must be validated by the Reflector admin team: users must be confirmed as subscribing before being activated.

You will receive a mail from the service confirming the details of your account request and then a second mail once the account has been activated.
When the request is submitted a confirmation page is displayed:

Request Submitted
Your account request has been forwarded to our admin team, please allow us some time to process it.

You will receive a confirmation e-mail once your account has been activated and is ready for use.

Thank you.
The first time you log in there will be no information to display:

![Image](https://example.com/image.png)

### 3.2 Create a project

To get audio streaming from source to destination using the Reflector service, you need to first set up a project to define the link. Allocating devices to a project allows you to set up an STL or multi destination audio distribution.

To set up multiple STLS/distributions, simply add a project for each.

Click on the '+ Add New' link brings you to the Create Project screen.

- give the project a name (mandatory) and description (optional)
When you click on create, the system will move to the 'Device List' screen, allowing you to start defining the component devices for the project.

Click on 'add new' to add a device to the project. To work, each project must have as a minimum an encoder and a decoder.
Enter the

- **Device name** – a label to recognise the device

- **MAC address**: this is unique to each device and is used to identify it to the system: Reflector Clients that call in and do not have a MAC address known to the system are ignored.

- **Type**: Devices either are encoders: the source (studio end) of the audio, or Decoders: the destination (transmitter/affiliate etc). Select which function you want this device to have. The later configuration options will change, depending upon the device type selected:

  **Encoder configuration options:**
  
  - **Audio input source**: which audio input is being used. On the Exstreamer 1000 when using the digital input, select SPDIF Optical / AES-EBU. Other options are self explanatory.
  
  - **Input gain**: the audio input can be amplified, adjust as necessary for your audio equipment, to obtain best results.
  
  - **Quality**: 5 quality options are offered, choose the one that matches your requirements (mono/stereo, quality). The higher the stream rate (in kbps) the higher the quality.
  
  - **Use relay one to**: Devices with relay outputs can have the device signal stream failure on Relay 1, or the relay can be set unused.

  **NOTE**: this relay flagging takes up to 90 seconds in Encoder mode

  **Decoder configuration options:**
  
  - **Buffer**: the decoder stores (buffers) the incoming audio before playing it back; this is to allow for the variable timing of the delivery of the audio over the
network. This buffering introduces a delay (latency) between the sent and received audio. Choose the buffering value that best matches your needs for latency and quality (too small a buffer can lead to audio artefacts caused by the network 'jitter' (delay in delivery of the audio)

**NOTE:** Using a buffer setting of 100ms will automatically disable the flagging of Stream Loss on local relay 1.

- Output level: the audio output can be amplified, adjust as necessary for your audio equipment, to obtain best results.
- Use relay one to: Local relay one can be used to either automatically signal stream loss, or show the value of the Project's Encoder Input 1.

Hardware devices with more than one relay output will automatically have their other relay outputs set to the corresponding Encoder Input (2>2, 3>3 and so on).

- Save as default: subsequent devices added to the project will have the same device type as this device suggested as the default value.

When you have selected your configuration options for your first device, click on 'Create'. This creates the Project with one device associated.

![UI screenshot](image)

The UI returns to the Devices List, showing the details of the device. Clicking on the “i” information symbol will display the configuration options.

Before the device makes contact with the Configuration Server, it will show up in red (error) with ‘No Contact’ and ‘No Status’, there will also be no IP address displayed, as the client informs the server of it's IP address.

Click on 'add new' to add further devices to the project.

Note the 'Project Data' tab – use this to view/edit the Project name and description.
3.3 **Home page**

Note the Green status overview: showing 'OK' for normal operation. Select a project to see more detail by clicking on the 'Modify' button.
3.4 Viewing project status

If you need to check on the status of the project at any time, go to the Reflector web UI and select the project, you will see the details of when the devices last contacted the Configuration service (according to the time zone of the server). The Clients should contact every 30s, so it will quickly be obvious if there is a communications problem:

Clicking on 'Cancel' returns you to the Home page with a list of your projects.

Clicking on the 'Delete' button next to a device will, after a confirmation dialogue, delete the device from the project.

Clicking on the 'Modify' button for a device will allow you to make changes to that device's settings: see the chapter “Modify device settings” for more detail.

Clicking on will display the configuration settings for the relevant device, and clicking on the will restore the original view:
3.5 Modify device settings

Clicking on the 'Modify' button next to a device shown in the Project view causes the details shown here to be displayed:

All of the device settings can be changed as required. Press 'Save' to update the details, or 'Cancel' to abort.
3.6 Modifying Account Details

The stored login details can be changed by clicking on the 'Modify your data' link in the left hand menu, the UI shows the screen below:

![Account Details Screen]

To change the password, it it necessary to enter the current password as well as the desired new one.

Optionally, you can have the system email the details to you for future reference.

3.7 Menu options

The Reflector Web UI maintains a standard Menu allowing the user instant access to a number of functions:

3.7.1 Home

Selecting this option returns the user to the list of their projects.

3.7.2 Projects

- List: Selecting this option returns the user to the list of their projects
• Add Project: clicking on this tab allows the user to define a new Reflector project using the functionality explained in the section Error: Reference source not found

3.7.3 Account Details
Using this option the user can update the registered details of their account.

3.7.4 Search

The search for an individual device can be done either by IP address, or by MAC address and the function also works on partial entries – entering “.247” and hitting <enter> or clicking on “search” will bring up all of the user's devices that have this string in their IP address, for instance.

Search by Project name works in the same way as search by MAC / IP.
4 Legal Information

© 2012 Barix AG, Zürich, Switzerland.
All rights reserved. All information is subject to change without notice.
All mentioned trademarks belong to their respective owners and are used for reference only.

Document version 2.04
July 2012
Documents Reflector Client firmware version 3.01 and Reflector Server UI version 0203.