AIRTIME

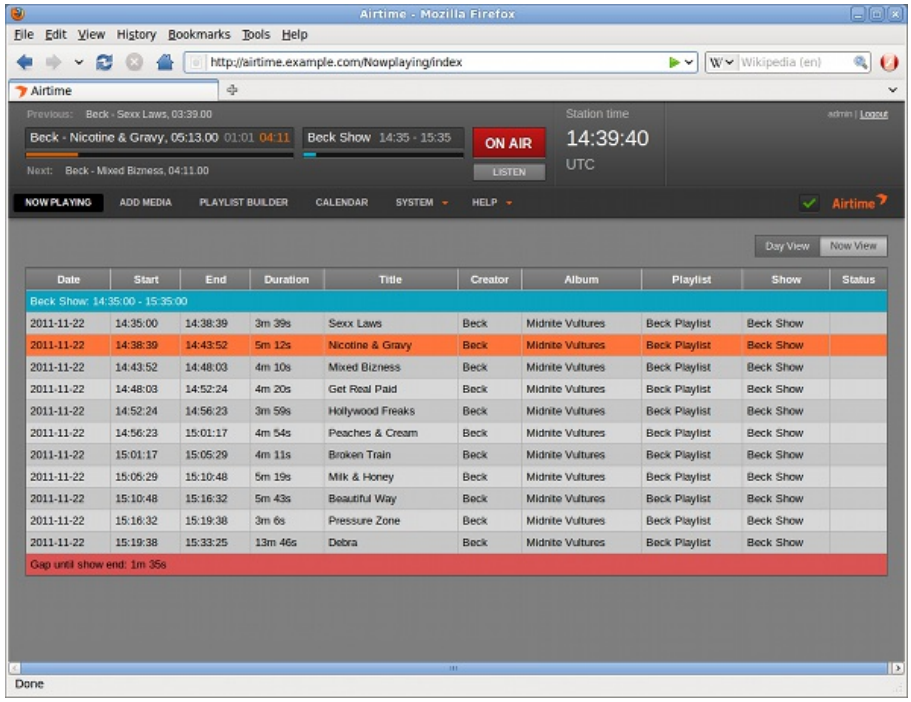
Published : 2012-01-23
License : None

INTRODUCTION

1. WHAT IS AIRTIME?

Updated for Airtime 2.0.0

Airtime is the open broadcast software for scheduling and remote station management. Web browser access to the station's media archive, multi-file upload and automatic metadata verification features are coupled with a collaborative on-line scheduling calendar and playlist management. The scheduling calendar is managed through an easy-to-use interface and triggers playout with sub-second precision.



Airtime has been intended to provide a solution for a wide range of broadcast projects, from community to public and commercial stations. The scalability of Airtime allows implementation in a number of scenarios, ranging from an unmanned broadcast unit accessed remotely through the Internet, to a local network of machines accessing a central Airtime storage system. Airtime supports the playout of files in both the commonly used MP3 format and the open, royalty-free equivalent Ogg Vorbis.

Airtime manages the Liquidsoap stream generator, which is at the heart of the system. Liquidsoap generates streams from files in the Airtime media archive, which is indexed in a PostgreSQL database. Live shows are automatically recorded with Ecasound, using the soundcard line input. Editors and station controllers can use Airtime to build playlists and manage files (upload, add metadata, manage advertisements) inside the station or via the Internet, using a standard web browser such as Mozilla Firefox.

The scheduler in Airtime has a calendar view, organized by months, weeks and days. Here the program editors can schedule playlists and shows for their broadcast station. In some scenarios, the transmitter is situated outside the reach of the broadcaster and all program management has to be maintained through the web interface. Possible reasons for this scenario might be of a pragmatic nature (running many stations from one central office due to limited human resources) or an emergency (running a transmitter in a crisis area without putting staff at risk).

AIRTIME WORKFLOW

This typical workflow is intended to clarify the difference between the various components that make up a complete Airtime system.

1. There are media files on a server, which include metadata in their tags (title, creator, genre and so on).

2. There is a PostgreSQL database which contains the location of those media files and their metadata. This means you can search for and select a set of media files according to the specific metadata that you require.

3. There is a Liquidsoap stream generator on the server which can take individual media files and assemble them into a continuous stream. This stream can be sent to a soundcard (e.g. for a broadcast mixer, on the way to an FM or DAB transmitter) or to a streaming server for IP network distribution, such as LAN, local WiFi or the Internet. You can stream to a soundcard and to IP networks with the same server, if you wish.

4. Icecast (http://www.icecast.org/) is the default streaming server, and there is also support for SHOUTcast (http://www.shoutcast.com), but in theory you could stream from Liquidsoap to any online service. If a suitable software interface is not available for your streaming service of choice, you can send audio from Liquidsoap to a separate encoding or streaming machine via a soundcard.

5. Airtime manages all of these components, and provides an easy to use web interface to the system. It enables your station staff to:

a) upload media files to the storage server

b) automatically import the file metadata into the PostgreSQL database

c) edit the metadata for the files, if required

d) create and edit playlists of media files (playlists are also saved in the database)

e) schedule shows (which can contain playlists, or be live) for specific dates and times on a calendar

f) record live shows in 256 kbps Ogg Vorbis format from the soundcard input with Ecasound, upload them to the storage server and import them into the database automatically

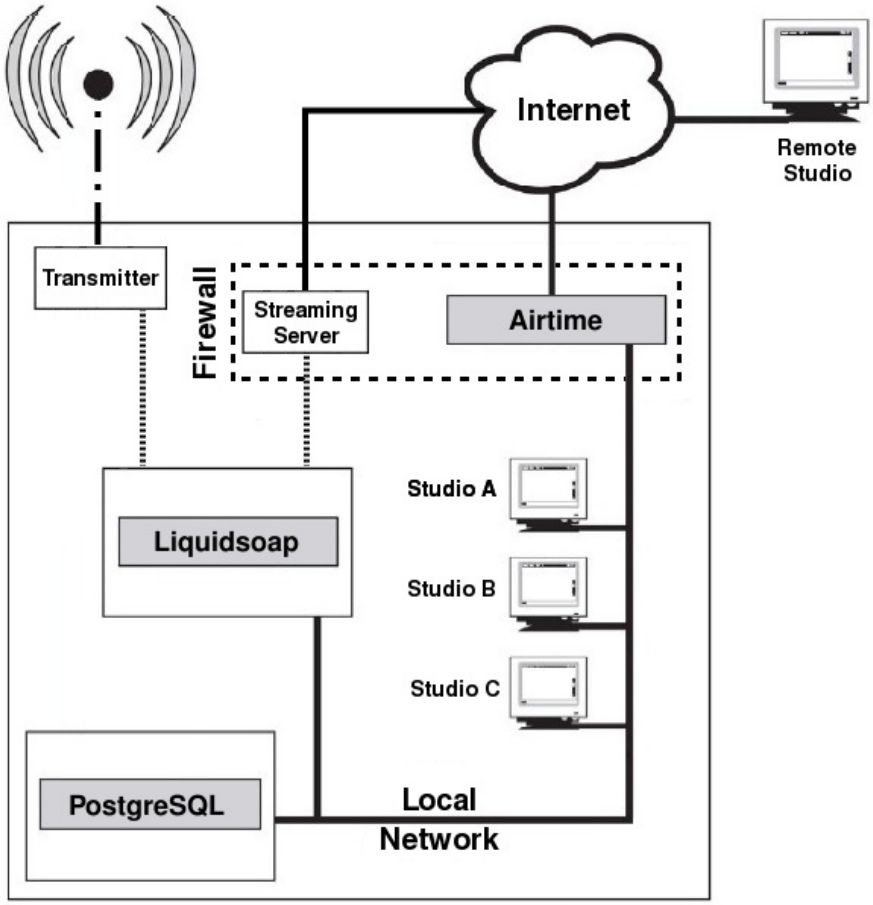
g) manage presenter access to the schedule calendar

h) see what is about to be played by Liquidsoap in the 'Now Playing' view

EXAMPLE SYSTEM

Combining Airtime, the Liquidsoap stream generator, the PostgreSQL database and file storage, you can supply a broadcast station with all the functionality needed for automation.

In the diagram below, the media files are stored on a separate machine which also includes a PostgreSQL database, accessible through the local network. Liquidsoap outputs streams to both the transmitter and a streaming media server. The machine running Airtime is behind a firewall because it is connected both to the local network, and to the Internet for remote access. This enables Airtime to offer password-protected access to the media database and scheduling from both inside and outside the studio building.


Internet
Remote
Studio
Transmitter
Firewall
Streaming
Server
Airtime
Studio A
Liquidsoap
Studio B
Studio C
PostgreSQL
Local
Network

2. RIGHTS AND ROYALTIES

If you're new to broadcasting, or have not streamed your station online before, reading the following brief explanation of compensation rules for musicians may save you a great deal of trouble later.

Independent music radio on the Internet is not what it might have been, due to royalty demands from SoundExchange in the USA, and similar organisations in other territories. These organisations are usually membership societies or government-sanctioned national authorities which are intended to collect money from broadcasters to compensate musicians for the use of their work. The royalty collection societies require payment before you can stream just about any music released on a commercial CD to the general public — whether you make any money out of streaming, or not. It's not so much the percentage of revenue demanded, but that there is usually an annual minimum fee to pay, which hurts small stations disproportionately.

For example, in the UK, the MCPS-PRS Limited Online Music Licence covers non-commercial music streaming by groups and individuals, as long as their gross revenue is less then £12,500 per year. The cost is on a sliding scale, up to £1,120 plus 20% tax per year for delivering up to 450,000 individual streams or serving 25,000 files; after that, you have to apply for a full MCPS-PRS Online Music Licence. That doesn't sound too bad at first, but 25,000 files per year works out at less than four downloads per hour for a round-the-clock website. This licence only covers publishing rights, not recording rights, so you have to negotiate an additional licence from Phonographic Performance Limited to actually play records or CDs.

Typically, you have to provide full statistical details to the royalty society of all music streamed or downloaded from your site. Even if your radio station is mostly speech, there are many limitations in the small print of these music licences. For instance, you can't use music for promotional purposes, and you can't stream a whole opera, without negotiating separate licences. Weirdly, you are not allowed to play a piece of music in a 'derogatory context' to the writer or performers; no drummer jokes allowed, then.

However, the biggest pitfall is that these MCPS-PRS licences only cover listeners in the UK. So if your Internet station picked up a significant number of listeners in other countries, you would have to pay for similar music licences in those countries as well. It's no wonder that many not-for-profit radio stations have disappeared from the virtual airwaves over the last few years, since not having the right licences could leave the operator liable to legal action. If you want to go down the paid licence route, and you can afford it, check out the http://www.prsformusic.com and http://www.ppluk.com websites for UK licence details. In the USA, the http://www.soundexchange.com website currently quotes a $500 minimum annual fee for non-commercial webcasters, plus a usage fee above a certain number of listener hours, for the right to stream music from its member record labels.

Free content streaming offers the chance that DIY Internet radio could rise again. Since royalty collection societies like MCPS-PRS and SoundExchange can only represent the interests of their own members, it follows that if you are not a member, you can stream your own self-produced content without paying for their licences. If you state somewhere on your website that the stream is of your own copyrighted material, and is made available to the public under a specific licence, then no-one should misunderstand your intentions. You might be able to persuade other people to allow you to stream their content too, as long as they do not have a conflicting legal obligation, such as having previously joined one of the many royalty collection societies around the world. You can ask for permission to stream when website visitors upload their own music files to you via a HTML form, much as the likes of MySpace do. Or you can collect files licensed under an appropriate Creative Commons (http://www.creativecommons.org) or other free content licence.

Explicit permission to stream on your particular server is always going to be the ideal, so think about your own terms and conditions before you accept files from third parties for streaming. How, for example, would you know if someone uploaded a file to your online radio station that unknown to you, had been ripped from a commercially released CD? That's the kind of thing that could get you in trouble with the licensing authorities and copyright holders.

USING AIRTIME

3. GETTING STARTED

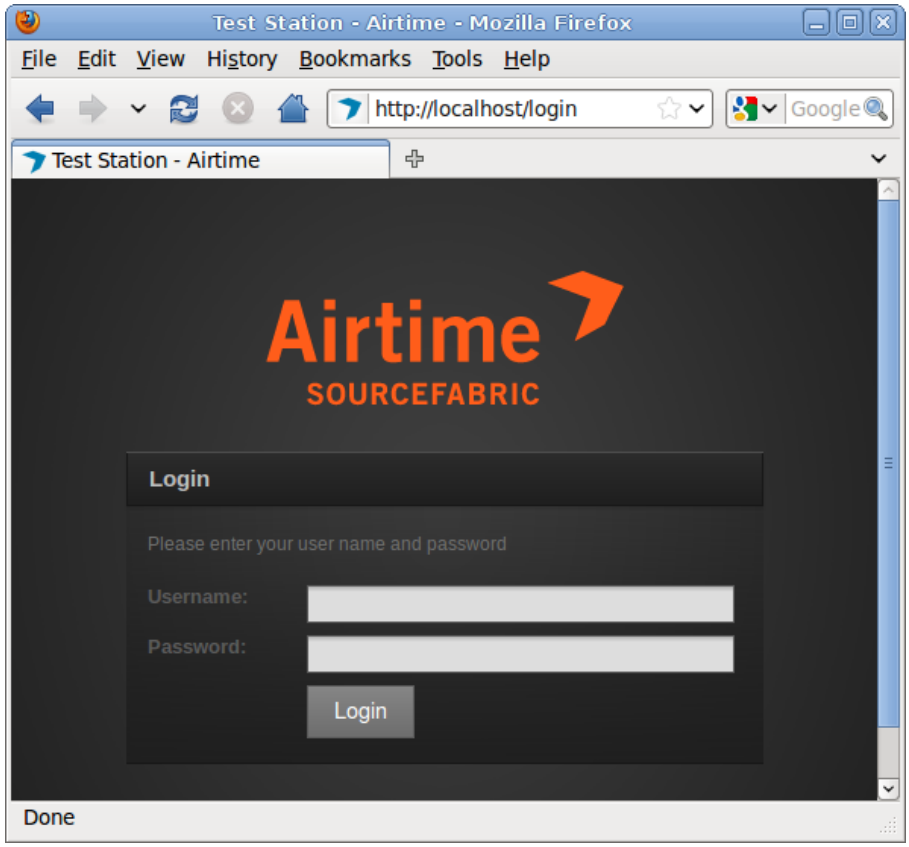
If the Airtime server has a web browser installed, you can access the administration interface by opening the address:

```
http://localhost/login
```

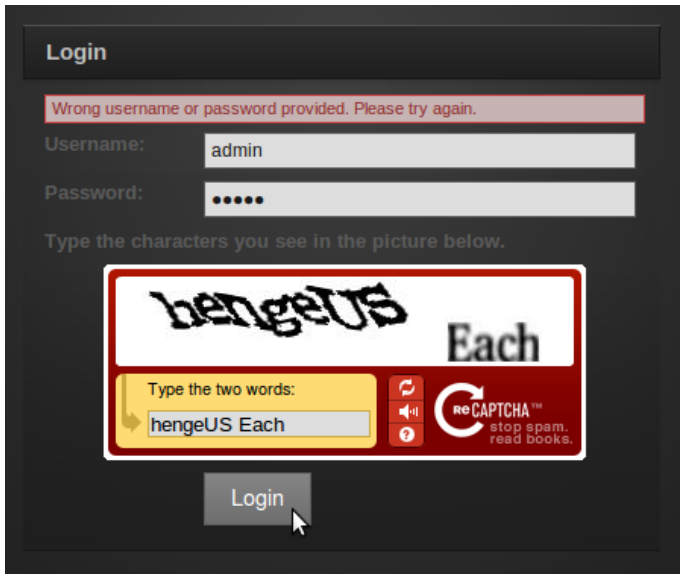
If you have set up Airtime so that it can be accessed from other computers, you would use a domain name instead. For example:

```
http://airtime.example.com/login
```

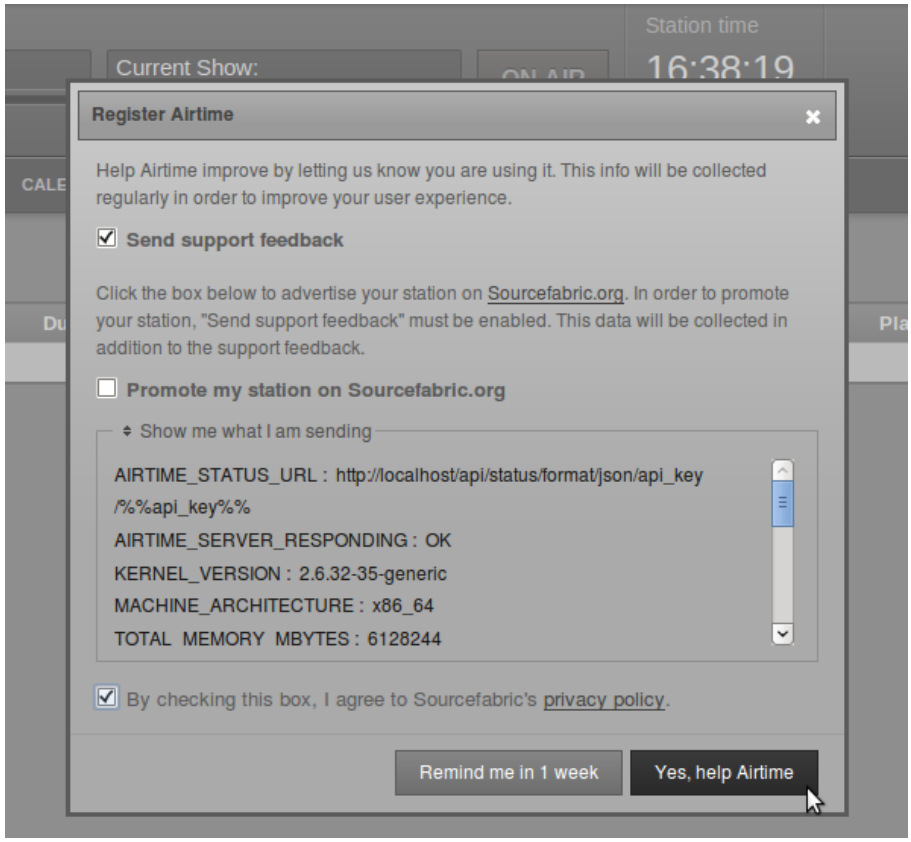
You can log in for the first time with the username *admin* and password *admin*. Your browser should automatically focus on the **Username** field.



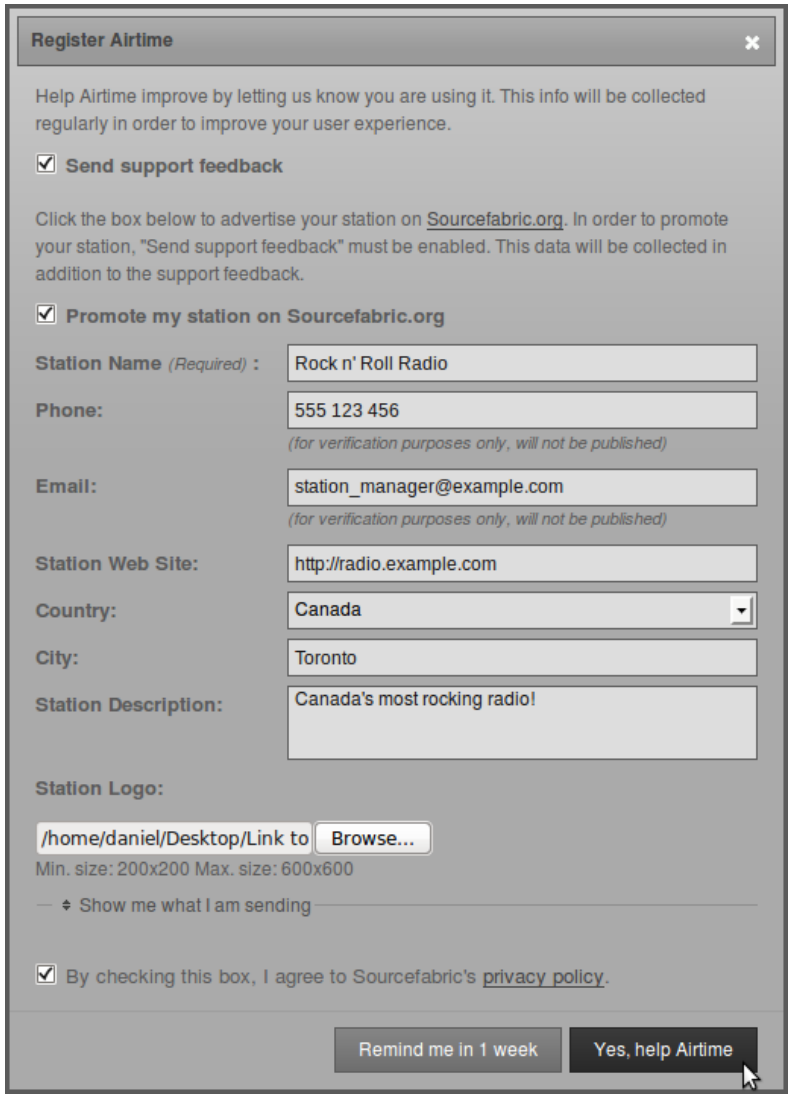
If you enter the password incorrectly three times, you will be presented with a reCAPTCHA challenge to prove that you are a human, and not a password-guessing robot. This feature helps protect your Airtime installation against brute force attacks.



After you have logged in as *admin* for the first time, a pop-up window will ask if you wish to send technical support data about your server to Sourcefabric. These details can be viewed by clicking on the **Show me what I am sending** link, which expands a scrolling window. The data helps Sourcefabric engineers resolve any problems with your Airtime installation, as well as count the number of installations worldwide. Sourcefabric has a privacy policy regarding data collection, which you can read by clicking the link to http://www.sourcefabric.org/en/about/policy/ further down. After checking the **Send support feedback** and **privacy policy** boxes, you can submit the data by clicking the **Yes, help Airtime** button.

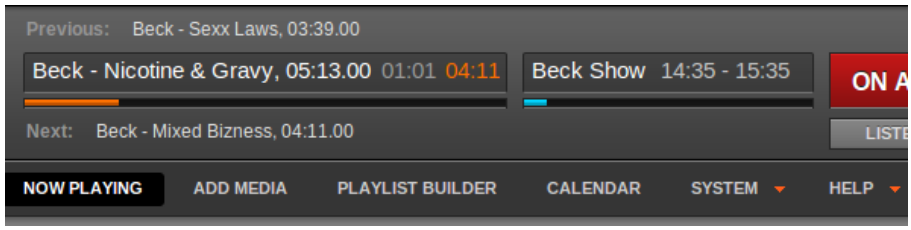


This window also offers the opportunity to **Promote my station on Sourcefabric.org** (on the page http://sourcefabric.org/en/products/airtime_whosusing) by checking the box. Fill in the form which will appear with some details about your station. The contact details are only requested for verification purposes, and will not be made available to the public. Click the **Browse** button to select a **Station Logo** image from the file manager on your computer.



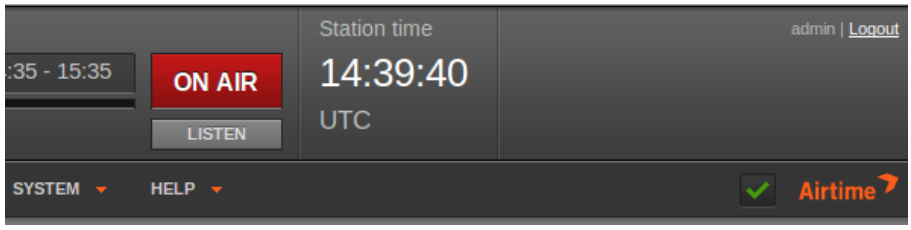
After the pop-up window is closed, you should now see the **Master Panel**, which is present at the top of every page of the Airtime interface. On the left hand side, the Master Panel displays the details of the **Previous** file played out, the current file playing (with an orange progress bar and time elapsed/time remaining), and the details of the **Next** file due to play. It also displays the name and scheduled time of the current show, with a blue progress bar.

Beneath this side of the Master Panel is the main navigation menu, with sub-menus labelled **Now Playing**, **Add Media**, **Playlist Builder**, **Calendar**, **System** and **Help**. We'll be looking at the contents of these menus in the following chapters of this book.



On the right hand side, the Master Panel features an **On Air** indicator, which turns from dark gray to red whenever a file is being played out. Underneath this is a **Listen** button which opens a pop-up player that can be used to audition playout streams. There is also a clock indicating the **Station time** and time zone.

In the top right corner, the name of the user currently logged in is displayed, and there is the link to **Logout**.



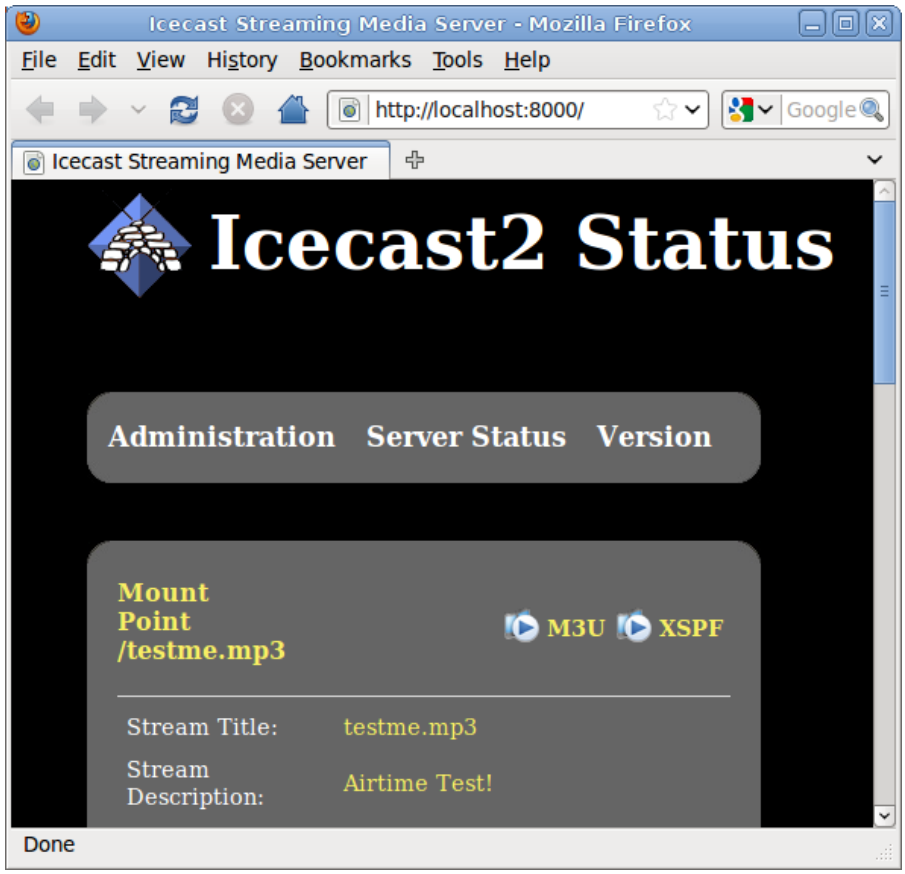
Over on the far right side, clicking on a green check mark opens a pop-up window with some information about the version of Airtime installed. If your Airtime installation is not the latest version available, the green check mark changes to a green upgrade arrow. Should your Airtime installation get too far out of date, this arrow will change to a red exclamation mark.

CHECKING AN ICECAST SERVER

If you have enabled the Icecast streaming option when you installed Airtime, you can check that the Icecast server is running by opening port 8000 of the server in your web browser. For example, on the server itself, you can use:

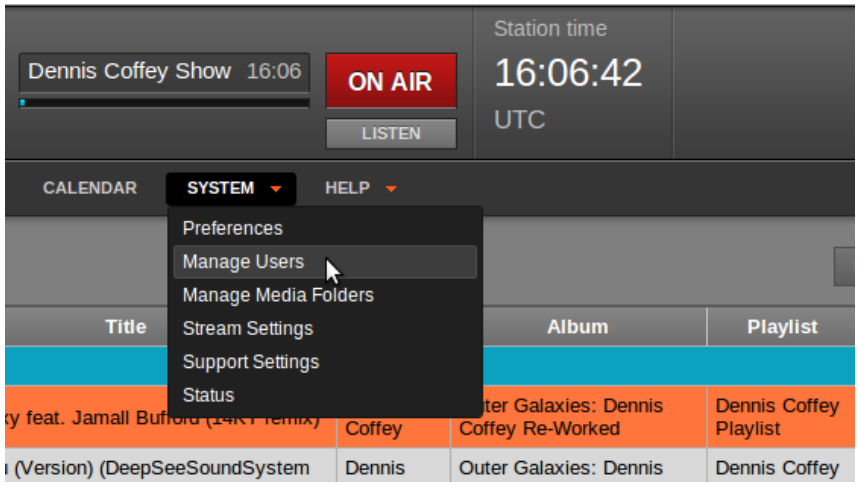
```
http://localhost:8000/
```

You should see the Icecast status page, with details of the connection that Airtime has made to Icecast. If you have only just installed Airtime, there won't be any media playing out yet.

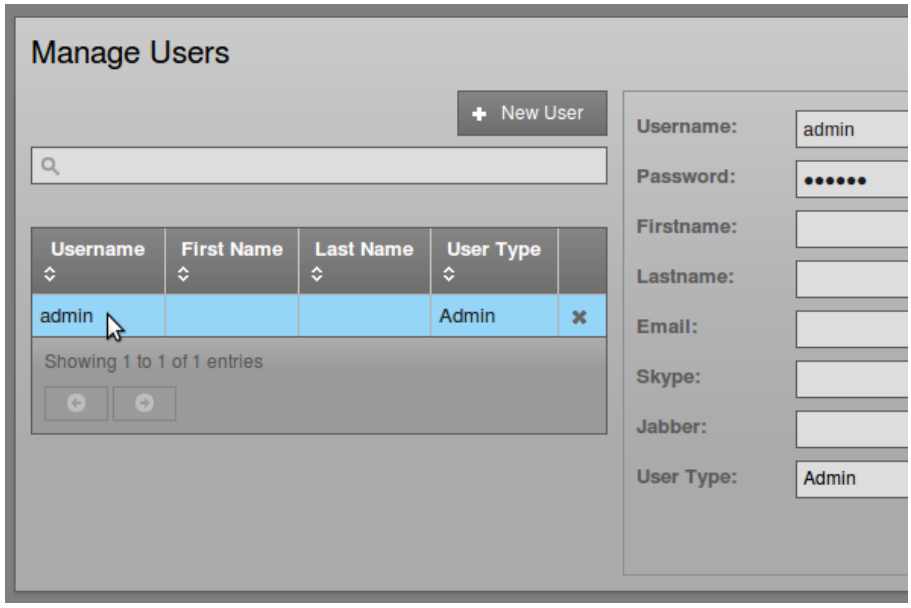

Icecast Streaming Media Server - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://localhost:8000/
Google
Icecast Streaming Media Server
Icecast2 Status
Administration Server Status Version
Mount Point /testme.mp3
M3U XSPF
Stream Title: testme.mp3
Stream Description: Airtime Test!
Done

4. MANAGE USERS

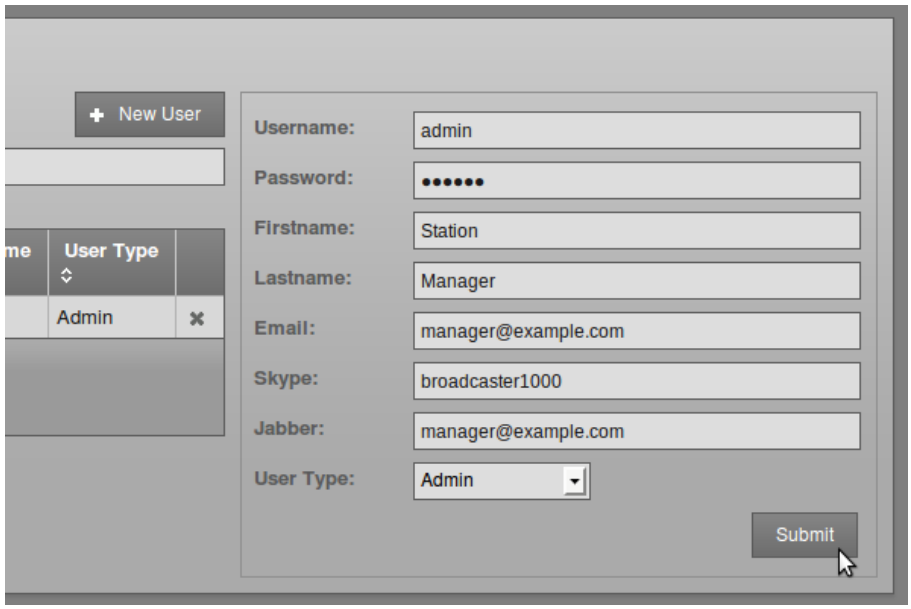
You should set a new administrator password immediately, using the **Manage Users** entry on the **System** menu.



On the left hand side of the **Manage Users** page, click on *admin* in the user list. The details for this user will appear in the box on the right hand side. At the moment, the *admin* user only has a **Username** and a **Password**.



Enter a secure password (as long and as varied as is practical) into the **Password** field. Keeping this password secure is essential for the smooth running of your station, because it enables access to all scheduling and management features. You can also enter other details for your user account on this page, including your full name and contact details. Then click the **Submit** button.



To add further user accounts to the system, one for each of your station staff that need access to Airtime, click the **New User** button. Enter a user name, password and contact details, and then select the **User Type** from the drop down menu, which can be *Admin*, *Program Manager*, *DJ*, or *Guest*. The difference between these user types is:

- An *Admin* (station manager) has read and write access to all the features of Airtime. This role should be reserved for trusted staff members only. If you give the *Admin* role to too many people, there could be arguments!
- A *Program Manager* has write access to the entire broadcast schedule, but cannot see the **Configure** menu and therefore cannot adjust Airtime preferences, manage user accounts or media folder settings.
- A *DJ* (presenter) only has write access to features related to the specific shows assigned to them by an *Admin* or *Program Manager*. The *DJ* can read data for other shows in the **Calendar**, but not write to them. This read access to other shows helps staff plan their own shows, for instance avoiding the scenario where two DJs play the same music by coincidence. Like a *Program Manager*, a *DJ* cannot see or use the **Configure** menu when they log in.
- A *Guest* can log in to read the forthcoming schedule or playlists, but has no write permission for any feature. *Guest* users cannot see the **Add Media**, **Playlist Builder** or **Configure** menus when they log in. The group of guest users might include the station accountant, or the show assistants.

New users that you add will be shown in the list on the left-hand side of the **Manage Users** page. If you have a large number of users on the system, you can use the search tool above the list (which has a magnifying glass icon) to identify specific user accounts. Click on the chevrons in the list headings to sort the search results by **Username**, **Firstname**, **Lastname** or **User Type**.

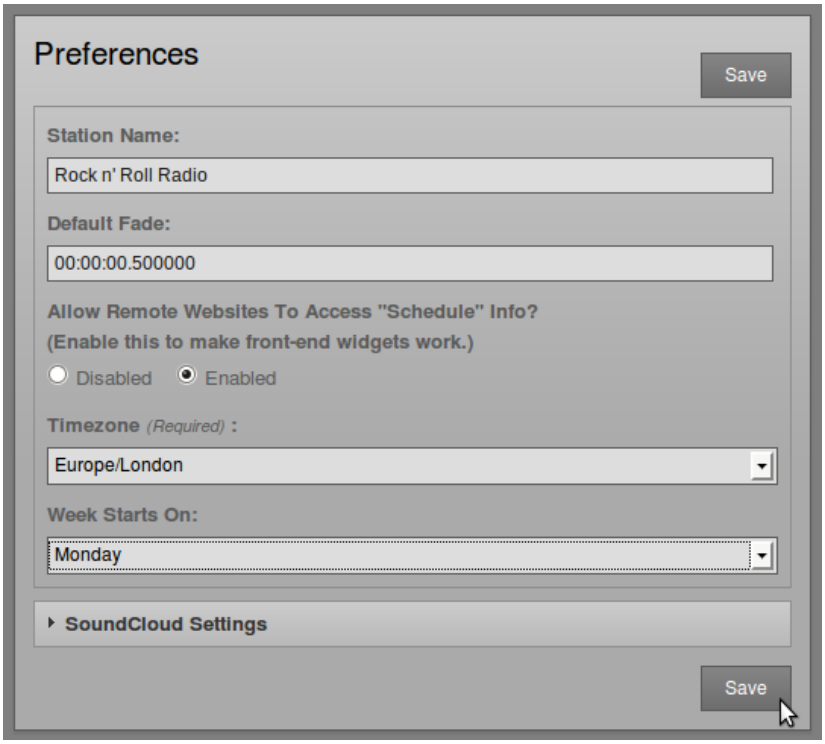
To edit a user account, click on that user's row in the list, change the user's details in the box on the right hand side, and then click the **Submit** button. To remove a user account, click the small **x** icon to the right side of its row in the list. You cannot delete your own user account.

5. PREFERENCES

On the **System** menu, click **Preferences** to set your **Station Name** and the **Default Fade** time that you would like to set for automated fades.

You can also enable live, read-only access to the Airtime schedule calendar for your station's public website with the **Allow Remote Websites to Access "Schedule" Info?** option, if you wish. (There is more about this feature in the *Exporting the schedule* chapter, in the *Advanced Configuration* section of this book).

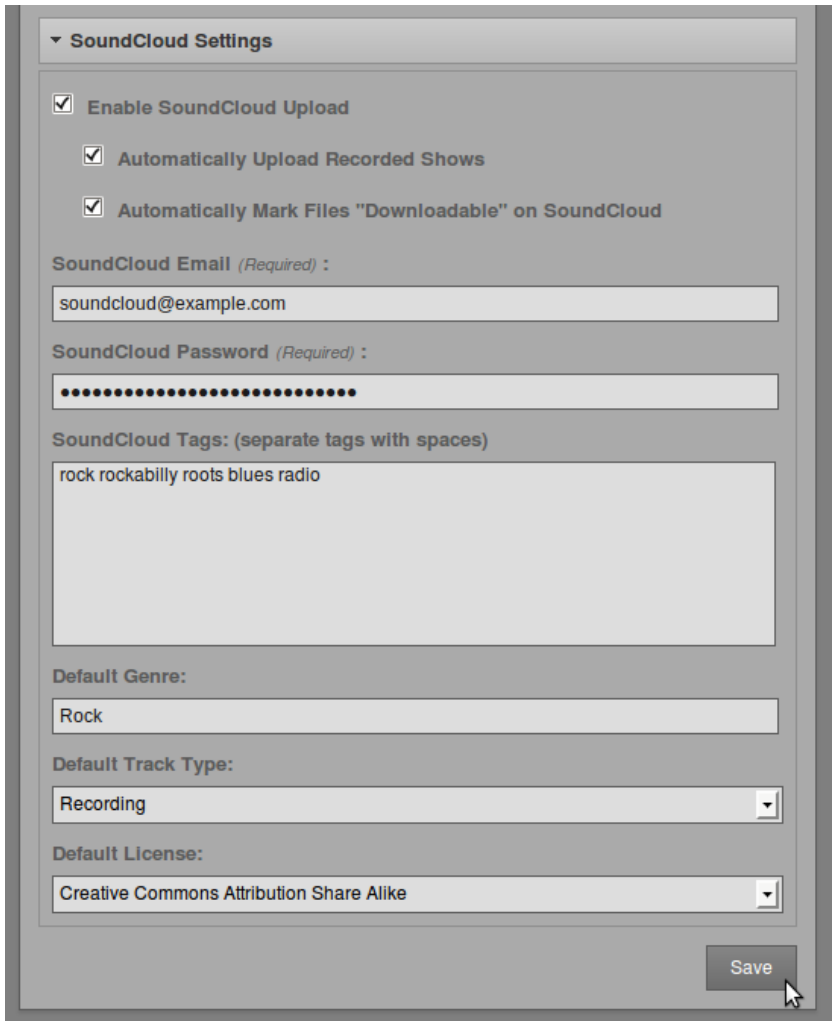
Use the **Timezone** drop-down menu to set local time at your station. Airtime stores show times internally in UTC format (*Greenwich Mean Time*), but can display local time for the convenience of your station staff. Also, you can set the day of the week that you wish to start your station's weekly schedule on, which defaults to Sunday. Then click the **Save** button.



SOUNDCLOUD SETTINGS

If your station has a SoundCloud account (on http://soundcloud.com), you may want to upload live show recordings to this service automatically. Click the small black triangle next to **SoundCloud Settings** to show this option. Apart from checking the **Enable Soundcloud Upload** and **Automatically Upload Recorded Shows** boxes, you will need to enter your SoundCloud login email address and password, and the tag metadata that SoundCloud will use to categorize your show recordings.

Check the box **Automatically Mark Files "Downloadable" on SoundCloud** if you wish to enable this option. You can also set a default genre, track type and copyright license here, including public domain, all rights reserved, or one of the *Creative Commons* licenses (see http://creativecommons.org). Then click the **Save** button again.



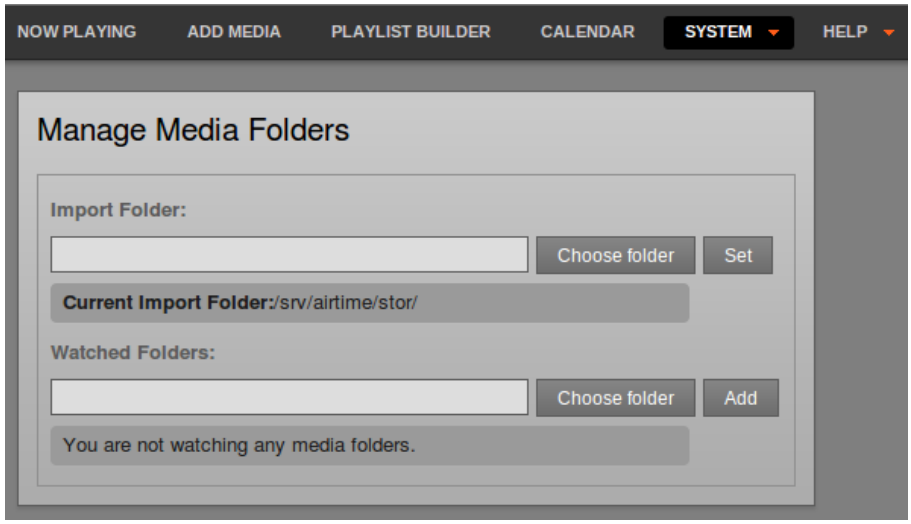
Please note that like most online distribution services, SoundCloud terms of service require you to have copyright in, or permission for Internet distribution from the copyright holder of, any media that you upload.

6. MANAGE MEDIA FOLDERS

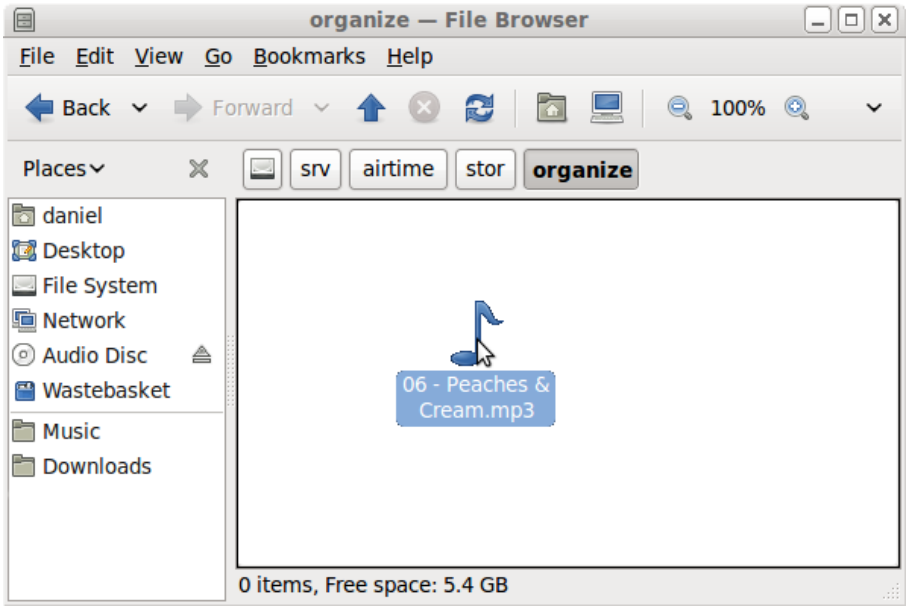
Airtime has a media storage archive, which by default is in the */srv/airtime/stor/* folder on your server. In **Manage media folders** on the **Configure** menu, you can change this default location, or add extra folders to be watched by Airtime.

You should not specify a network drive as a media folder, because if that network becomes disconnected for any reason, Airtime's **media monitor** will remove the missing files from its database. If those missing files are scheduled for a future show, the broadcast output for that show would be silenced.

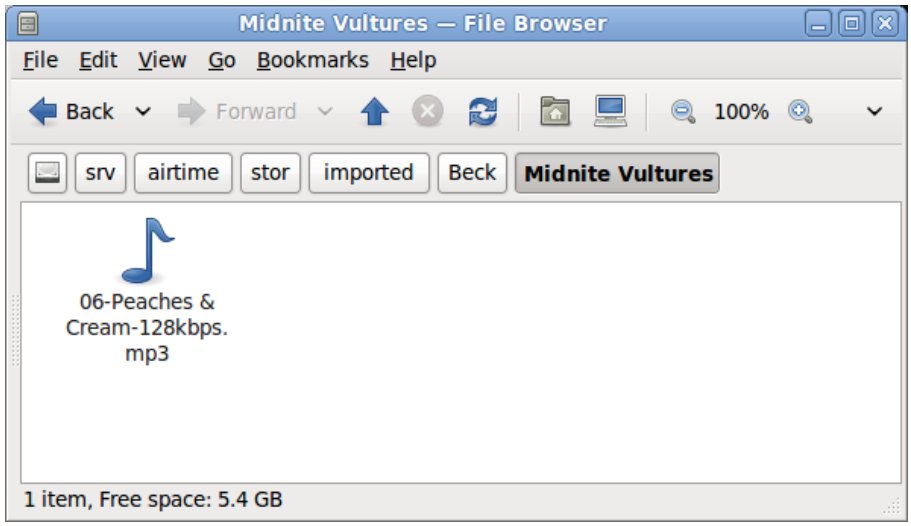
For the same reason, if a removable drive (such as a USB memory stick or MP3 player) is specified as a watched folder, that drive has to be present and powered on at all times. If your station staff use removable drives to store media files, it is much safer to use the **Add media** page of the Airtime administration interface or the **airtime-import copy** command to copy the files to the main storage server. See the chapters *Add media* and *Using the airtime-import script* for more details.



Any new media files you add to the */srv/airtime/stor/organize* folder or a watched folder will be automatically imported into the Airtime database. The *organize* or watched folders can be exported to computers on the local network, to enable dragging and dropping of media uploads using the file managers on the desktop computers at your studio.



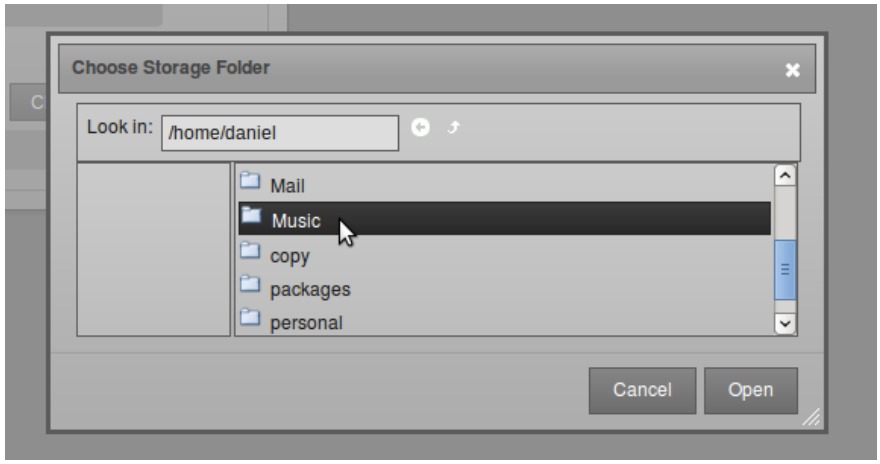
A file dropped into the *organize* folder will appear to vanish as the metadata is read, and the file is moved to the correct location in the storage hierarchy, according to its creator and title. This could be under a filesystem path such as */srv/airtime/stor/imported/Beck/Midnite Vultures/* in the screenshot below.



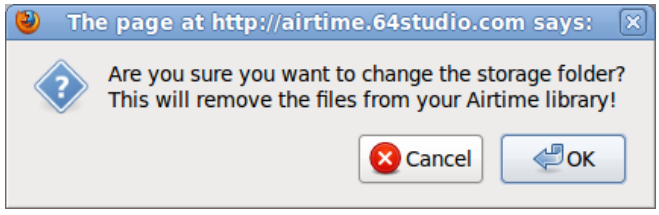
Files in watched folders are not moved into the main Airtime storage folder, but any files deleted from a watched folder will be automatically removed from the Airtime database.

CHANGING THE STORAGE FOLDER

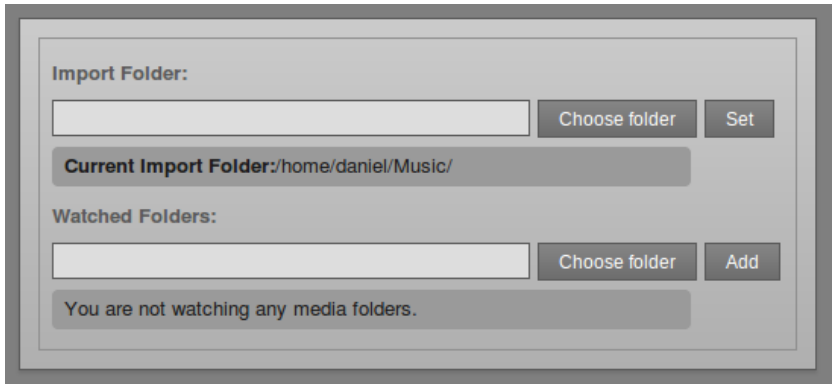
To change Airtime's storage folder, click the upper **choose folder** button. In the pop-up window that opens, double-click on the folder names to select the folder that you require. Then click the **Open** button to open that folder.



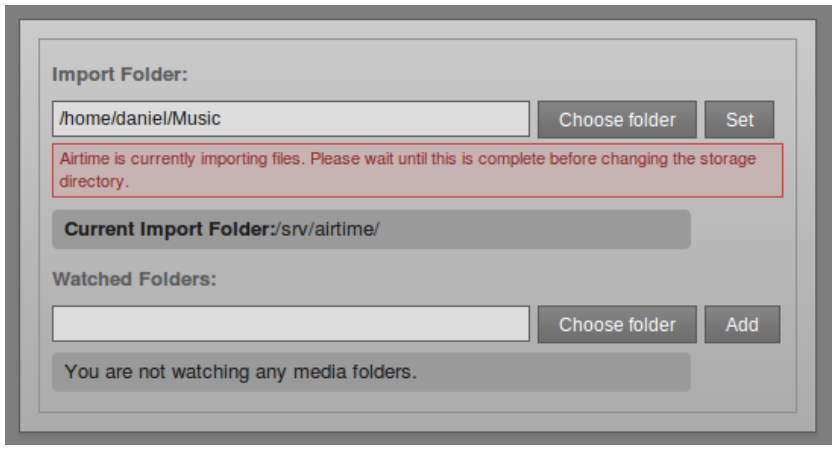
Back on the **Manage Media Folders** page, click the **Set** button to change the storage folder. Airtime will ask if you are sure about this action. Click the **OK** button to confirm your choice.



The **Manage Media Folders** page will now display the new storage location.

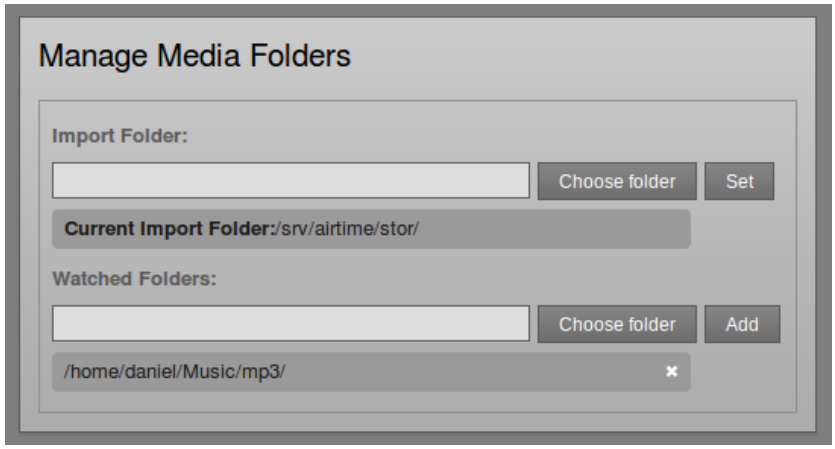


The storage folder cannot be changed while a file import is in progress. If you attempt to do this, an error message will be displayed.

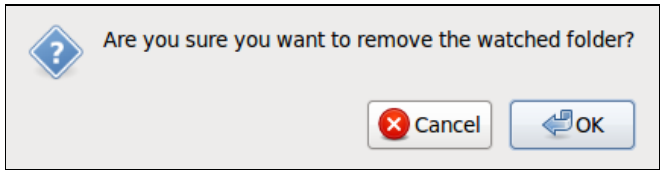


WATCHING A FOLDER

Under **Watched Folders**, click the lower **choose folder** button, open the folder you require, and then click the **Add** button. You can add as many watched folders as you require.



To remove a watched folder, click the small **x** on the right side of its row in the list. Again, you will be asked to confirm if you are sure about the action.

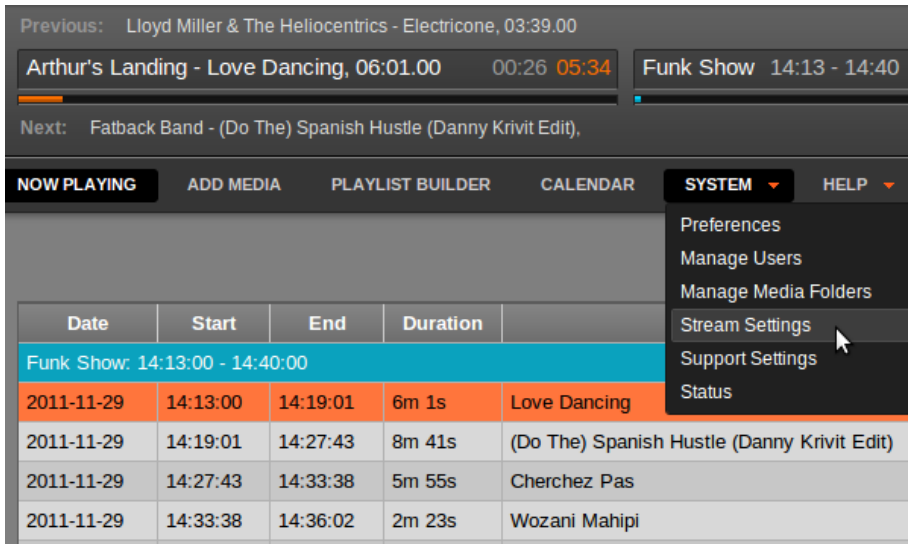


Edits to your stored and watched media files are automatically noticed by Airtime. If you edit any file in the archive, such as trimming the duration of a track, Airtime will automatically adjust the playlist and show lengths for that particular file.

The directory structure and file names in the storage archive are human-readable. This means you can find files for download and editing using a file browser on your server, as well as through the **Search** tab in the **Playlist Builder**.

7. STREAM SETTINGS

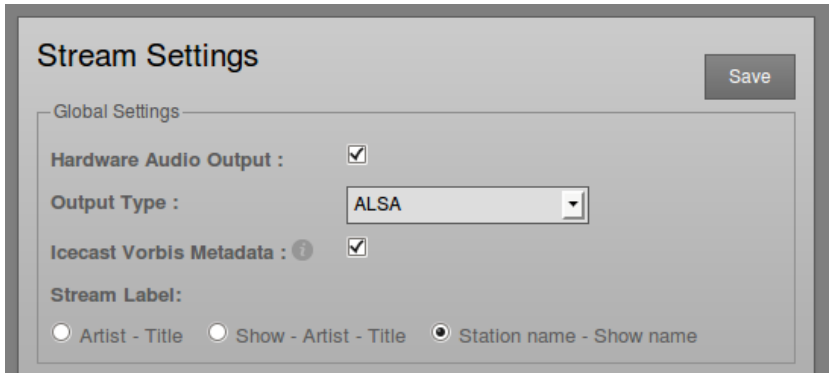
On the **System** menu, you can configure direct Icecast and SHOUTcast streams as well as soundcard output by going to the **Stream Settings** page.



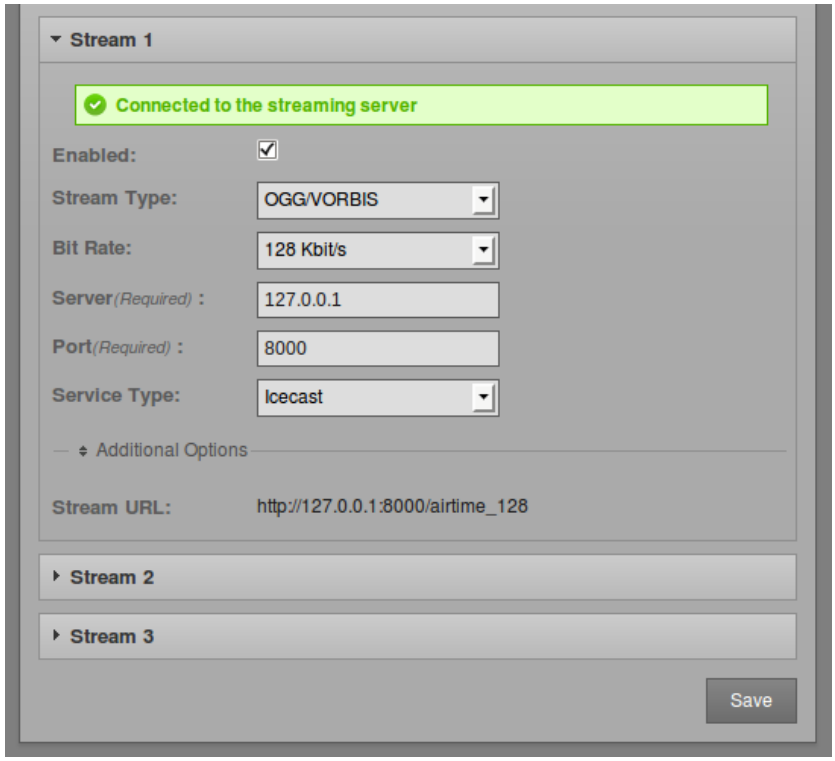
At the top of the **Stream Settings** page is a checkbox for global options including **Hardware Audio Output**, which enables playout from the default sound card on the server, if one is fitted. The default **Output Type** of *ALSA* on the drop-down menu will be suitable for most servers with a soundcard. If not, you have the option to choose from other Liquidsoap interfaces available, such as *OSS* or *PortAudio*. If you are only using the hardware audio output, and will not be streaming directly to Icecast or SHOUTcast, you can click the **Save** button at this point.

The second checkbox enables the sending of **Icecast Vorbis Metadata** with direct streams. This setting is optional because some media players have been known to disconnect from Ogg Vorbis streams when an Icecast server notifies the player that a new track is starting.

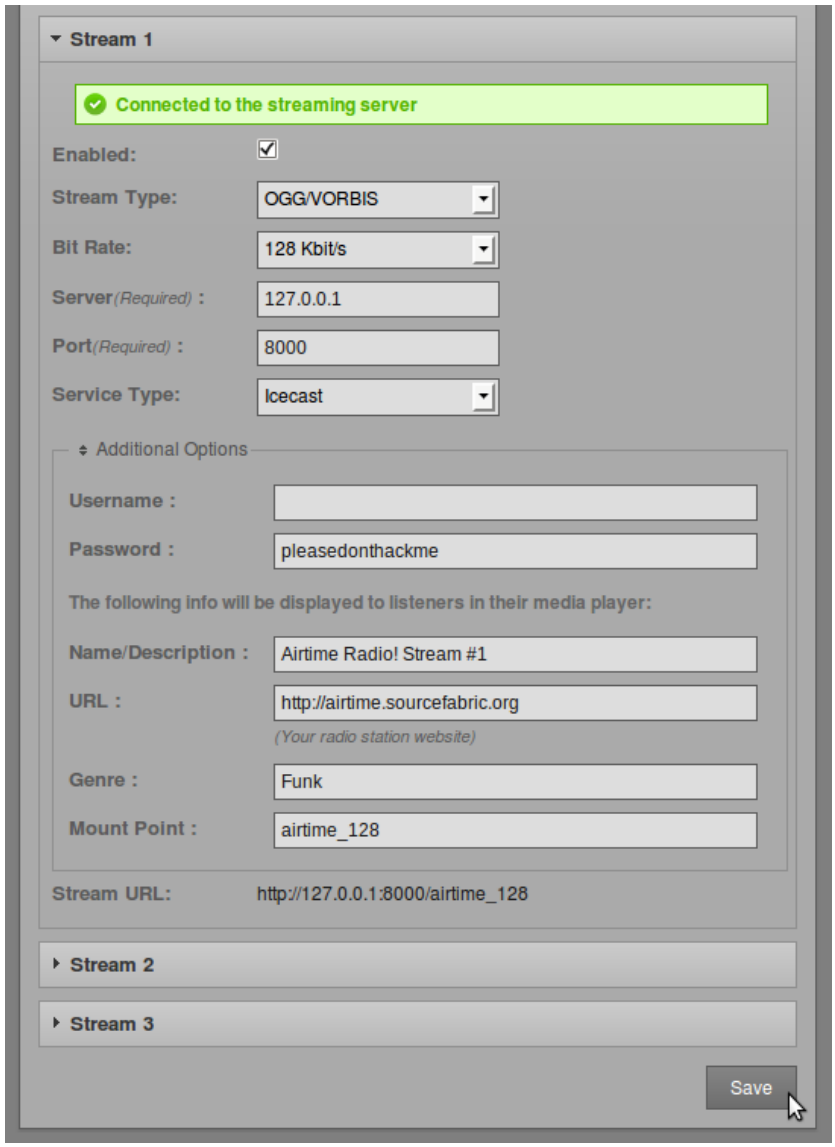
The **Stream Label** radio button allows you to set the metadata that will be sent with direct streams; *Artist* and *Title*, *Show*, *Artist* and *Title*, or *Station name* and *Show name*.



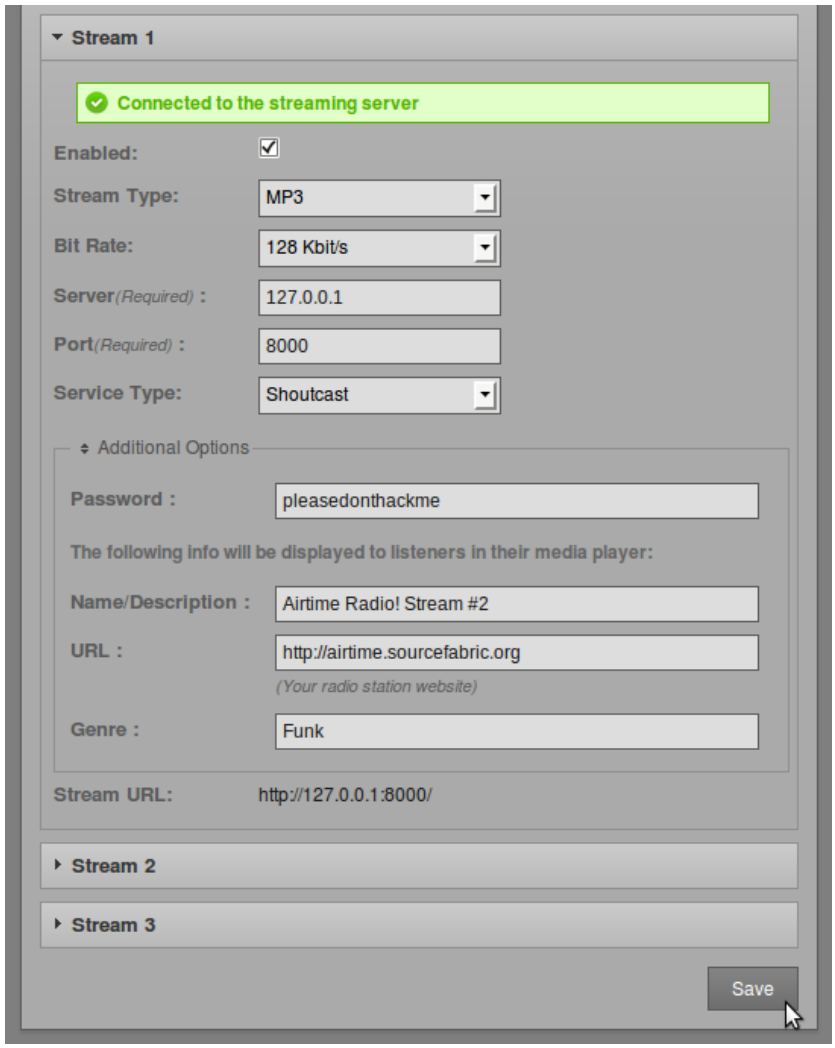
You can configure up to three independent streams with different bit rates, and send these streams to different Icecast or SHOUTcast servers. By default, only Stream 1 is enabled, with this stream being sent to Icecast on the same server at the *localhost* IP address of 127.0.0.1. To enable the other streams, click on the stream name to expand its box, click on the **Enabled** checkbox, and enter at least the **Server** IP address or domain name and **Port** details.



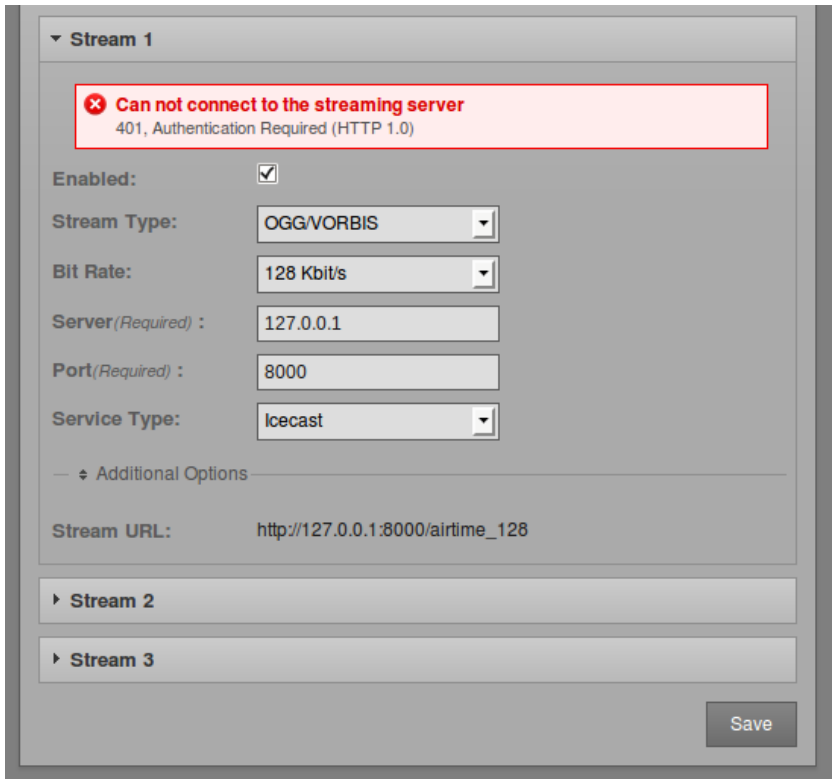
Click **Additional Options** to expand a box in which you can enter the **Username**, **Password** and metadata for the streaming server. You can also set the specific **Mount Point** that listeners will connect to here. Then click the **Save** button at the bottom right corner to update the Airtime server's settings.



When selecting a SHOUTcast server from the **Service Type** drop-down menu, you are restricted to using MP3 format only, so the choice of Ogg Vorbis format is grayed out in the **Stream Type** drop-down menu.



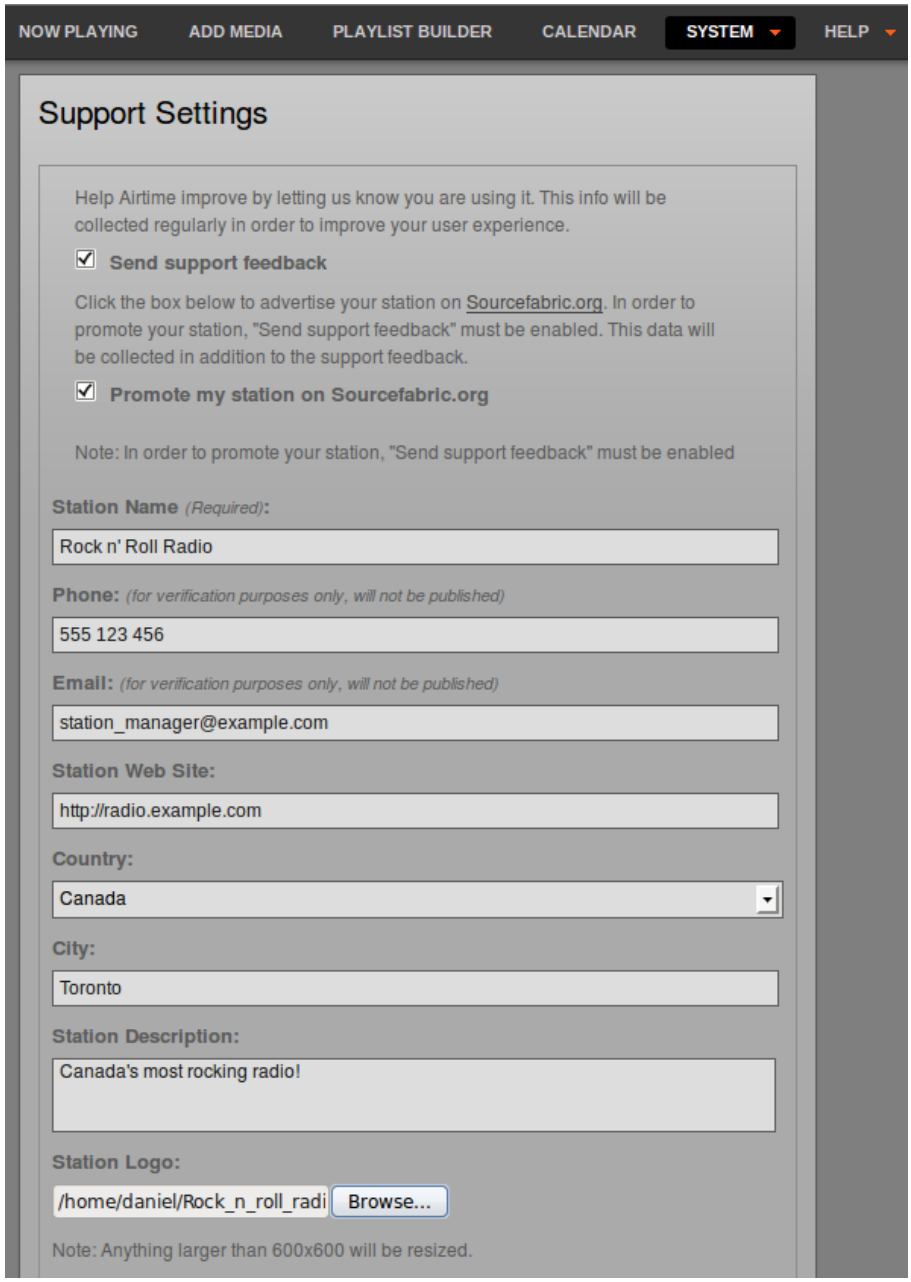
Any connection problems between Liquidsoap and Icecast or SHOUTcast are shown on this page. For example, if you enter the wrong **Password**, you will see an error message. To fix this, enter the correct password in the **Additional Options** box, and click the **Save** button.


Stream 1
Can not connect to the streaming server
401, Authentication Required (HTTP 1.0)
Enabled:
Stream Type:
OGG/VORBIS
Bit Rate:
128 Kbit/s
Server (Required) :
127.0.0.1
Port (Required) :
8000
Service Type:
Icecast
Additional Options
Stream URL:
http://127.0.0.1:8000/airtime_128
Stream 2
Stream 3
Save

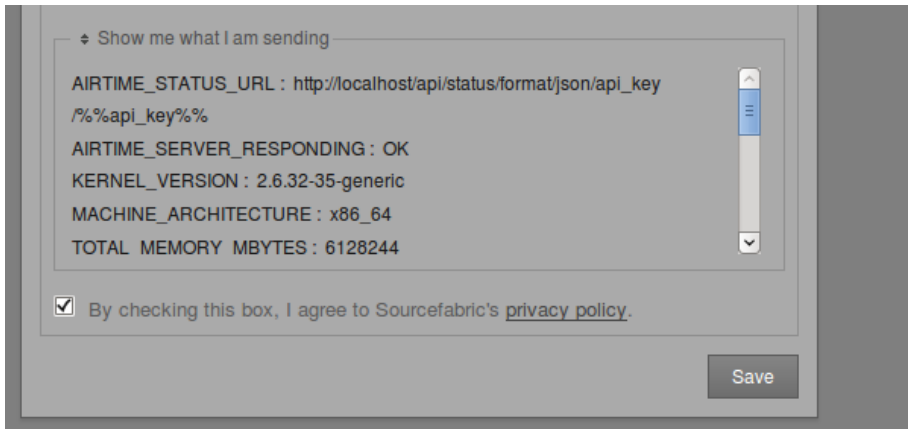
8. SUPPORT SETTINGS

If you did not already register Airtime when you installed it, as shown in the *Getting Started* chapter, you can click **Support Settings** on the **Configure** menu to display Airtime's automated feedback options. Check the **Send support feedback** box in order to post technical details about your Airtime installation to Sourcefabric, over the Internet. These details help Sourcefabric diagnose any problem that you might be having with your broadcast automation system.

You may also wish to send details of your station to Sourcefabric, so that your station has the opportunity to be promoted with other Airtime users on the http://www.sourcefabric.org website. This feature also helps Sourcefabric target its support services to the countries where they are needed most. Check the box **Promote my station on Sourcefabric.org** and fill in the details of your station. You can upload a station logo file from your desktop computer by clicking the **Browse** button.

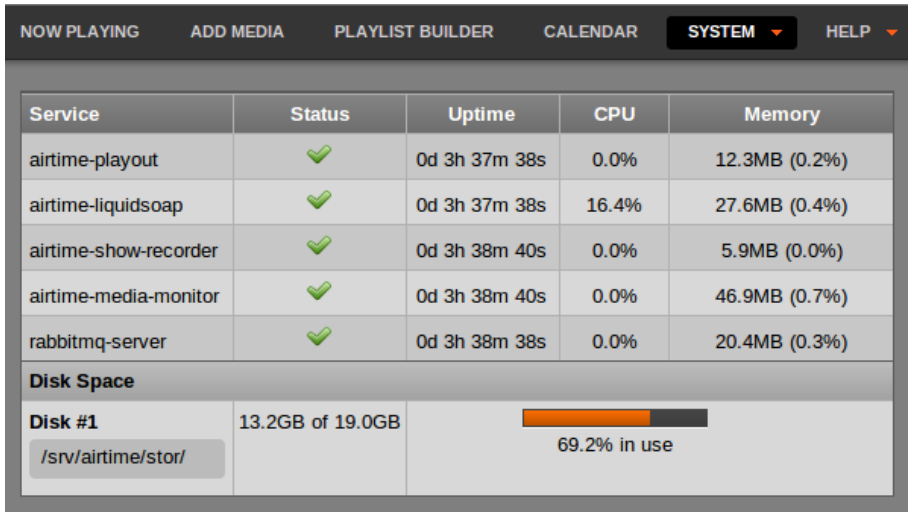


Click on the **Show me what I am sending** link to expand a box which displays the technical data being returned to Sourcefabric. The data is collected according to the Sourcefabric privacy policy (http://www.sourcefabric.org/en/about/policy/) which you are required to agree to before you can submit the information.


Show me what I am sending
AIRTIME_STATUS_URL : http://localhost/api/status/format/json/api_key
/%%api_key%%
AIRTIME_SERVER_RESPONDING : OK
KERNEL_VERSION : 2.6.32-35-generic
MACHINE_ARCHITECTURE : x86_64
TOTAL MEMORY MBYTES : 6128244
By checking this box, I agree to Sourcefabric's privacy policy.
Save

9. STATUS

On the **System** menu, the **Status** page provides an overview of the health and resource usage of the various services that make up an Airtime system. If all is well, you will only see green check mark icons in the **Status** column. This page also shows how much **Disk Space** you have used for media storage.

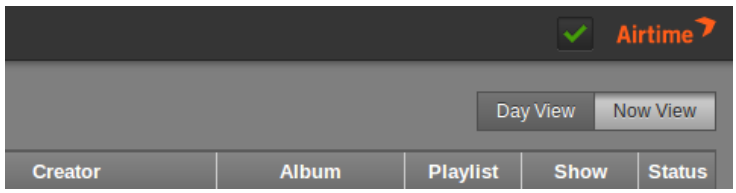


If any of the check mark icons in the **Status** column have changed to a red warning sign, contact your system administrator for assistance. Airtime will do its best to restart any failing services, but sometimes manual intervention may be required; for example, in the case of hardware failure.

If you have run out of storage space, an Airtime user with *admin* privileges could log in and delete media files that are no longer required from the **Playlist Builder**. Alternatively, you could move some files to a **watched folder** on another disk, or ask your system administrator to install additional storage capacity.

10. NOW PLAYING

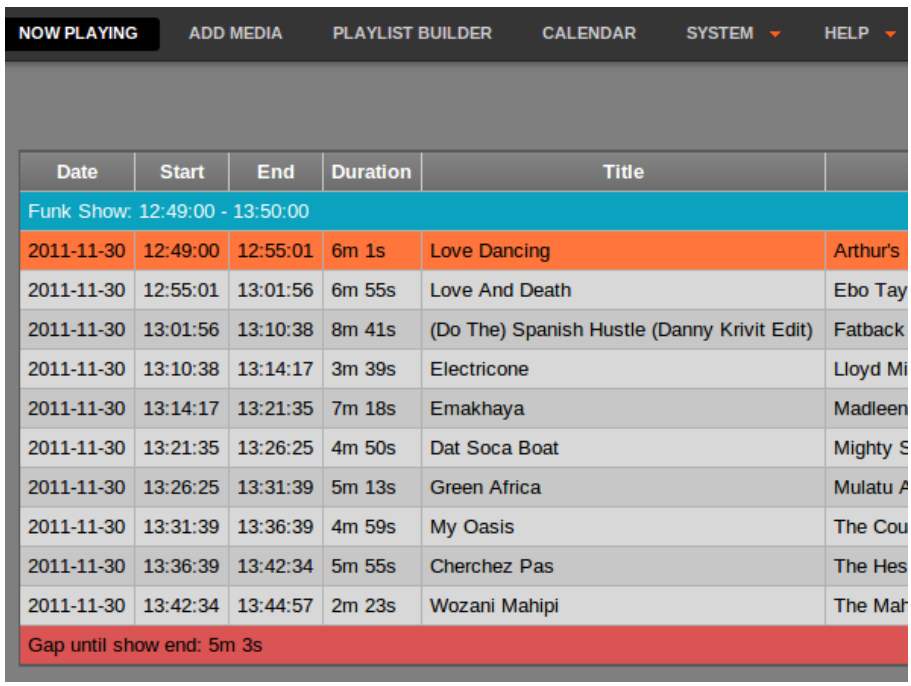
This page provides two views of the files your station is playing, which can be switched using the **Day View** and **Now View** buttons in the top right corner of the administration interface. If you've only just installed Airtime, there won't be any files shown in either view yet.



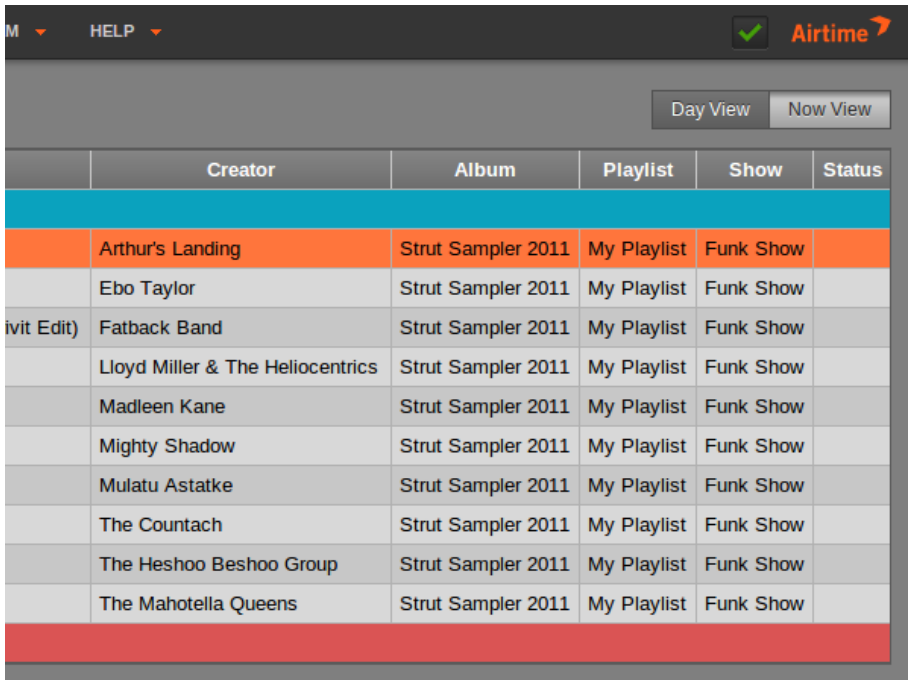
The default is **Now View**, in which files that have already been played are removed from the top of the list automatically. **Day View** shows all the files scheduled for playout today.

In both views, show titles have a blue background, while individual files have a light gray background. The row for the currently playing file is displayed with an orange background. Any underbooked shows (shows with insufficient content to fill the time allowed) are displayed with a row indicating the length of the gap in seconds. These **Gap until show end** rows have a red background.

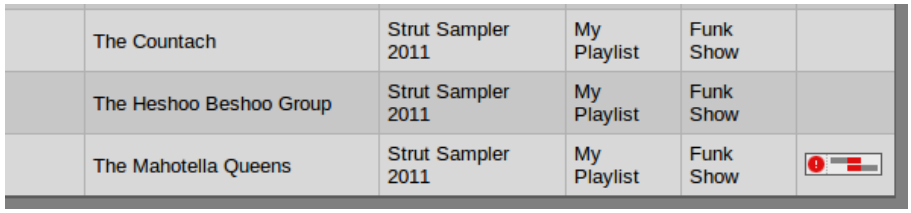
On the left side of the **Now Playing** page, the **Date**, **Start** and **End** times, **Duration** and **Title** of the file to be played are shown.



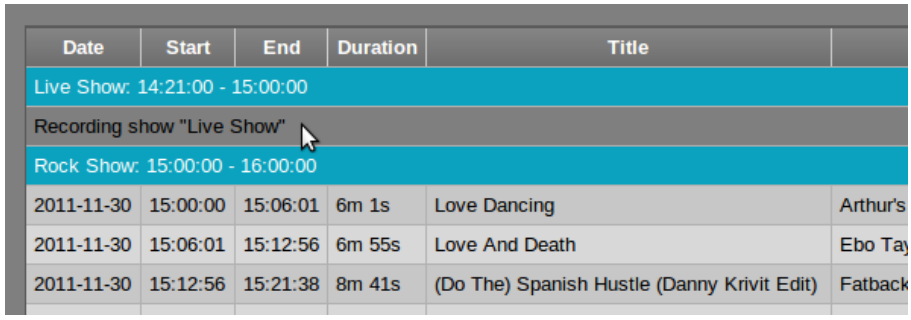
On the right hand side, the **Creator**, **Album**, **Playlist** and **Show** that each file is a part of is displayed. (Creating playlists and shows is covered in later chapters of this book).



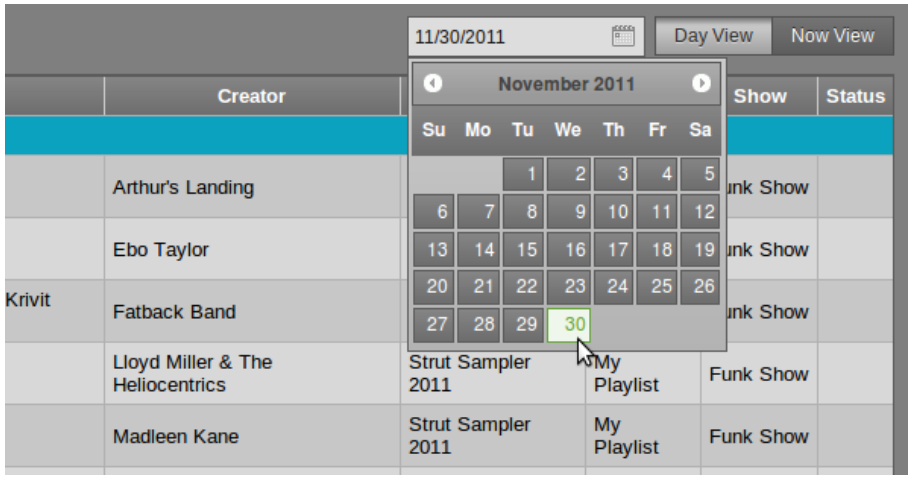
If a show is overbooked, which means the total playout length is longer than the time allowed for the show, a red and white warning icon is shown in the **Status** column. This warning icon indicates that the file in this row will be cut off when the show ends.



A live show that is set for recording is displayed with a dark gray background, to indicate that it does not contain any playlists.



The **Day View** looks the same as the **Now View**, except that files which have already played out are displayed at the top of the window. You can select which day of the schedule you wish to view by clicking on the date box, just to the left of the **Day View** button, and selecting a day from the pop-up calendar.

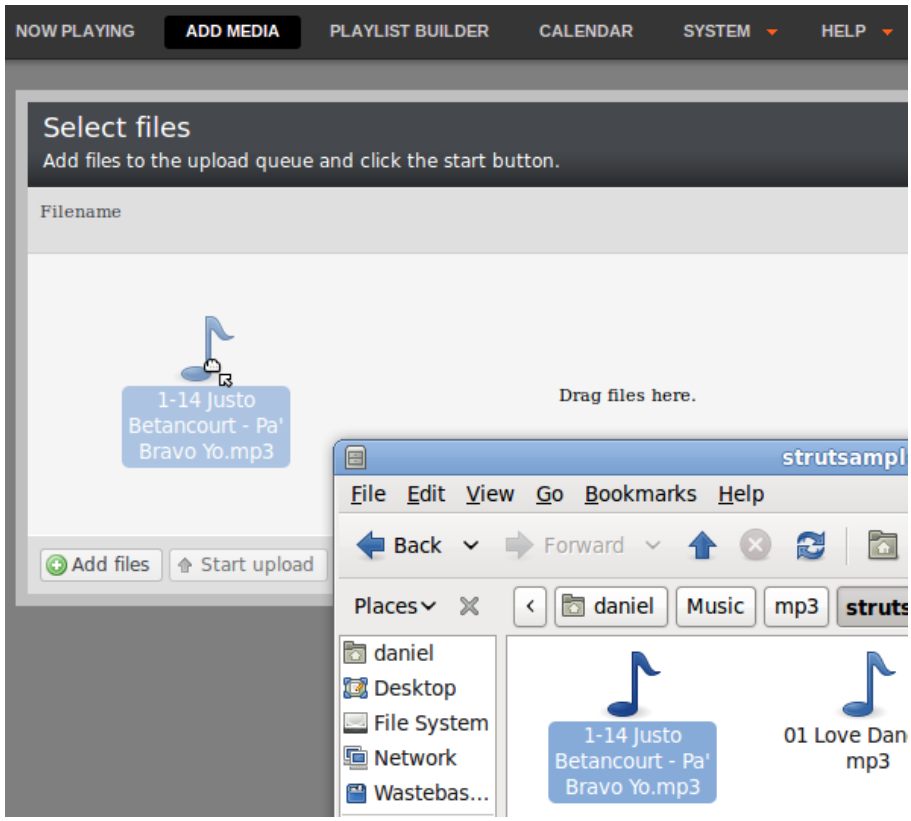

11/30/2011
Day View
Now View
November 2011
Su Mo Tu We Th Fr Sa
1 2 3 4 5
6 7 8 9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29 30
Creator
Show
Status
Arthur's Landing
Ebo Taylor
Krivit
Fatback Band
Lloyd Miller & The Heliocentrics
Strut Sampler 2011
My Playlist
Funk Show
Madleen Kane
Strut Sampler 2011
My Playlist
Funk Show

11. ADD MEDIA

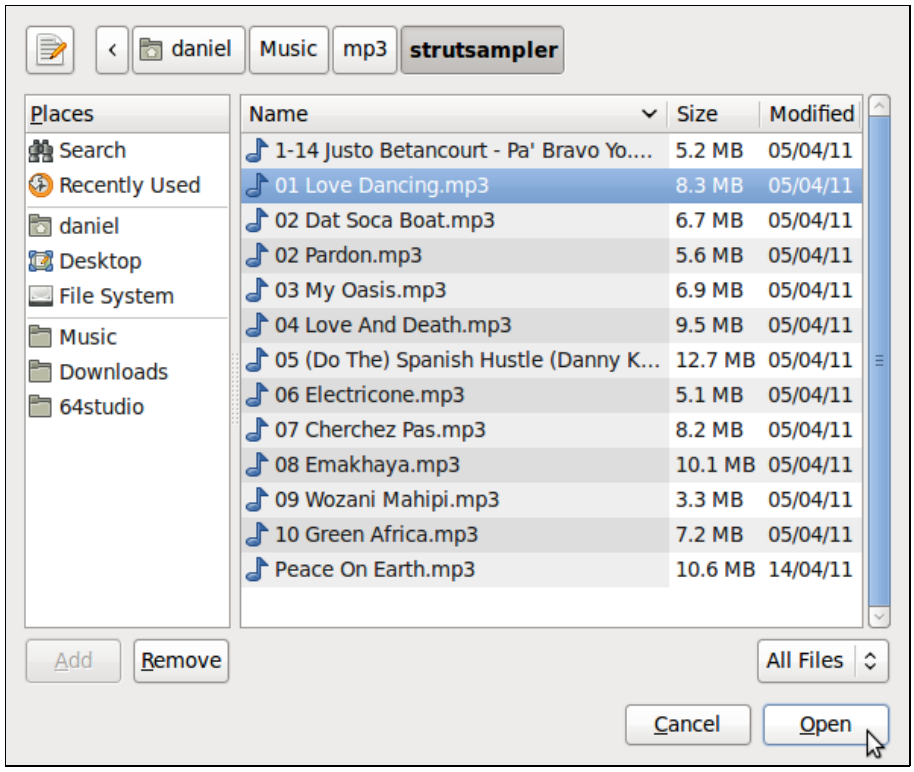
If you do not have direct access to the Airtime server, you can add files to the Airtime storage and database using the **Add media** page of the administration interface. This page includes an upload queue for media files, which supports drag and drop from your computer's file manager if you are using the **Mozilla Firefox 3.6** (or later) web browser.

On an Ubuntu Lucid desktop machine, Firefox supports uploading files to Airtime of up to 200MB in size. Other browsers and platforms may set an upload limit at 2GB. If you need to upload files larger than 200MB to the Airtime server, you may find it more convenient to perform the upload using SFTP, rather than through the browser. See the chapter *Automated file import* for more details.

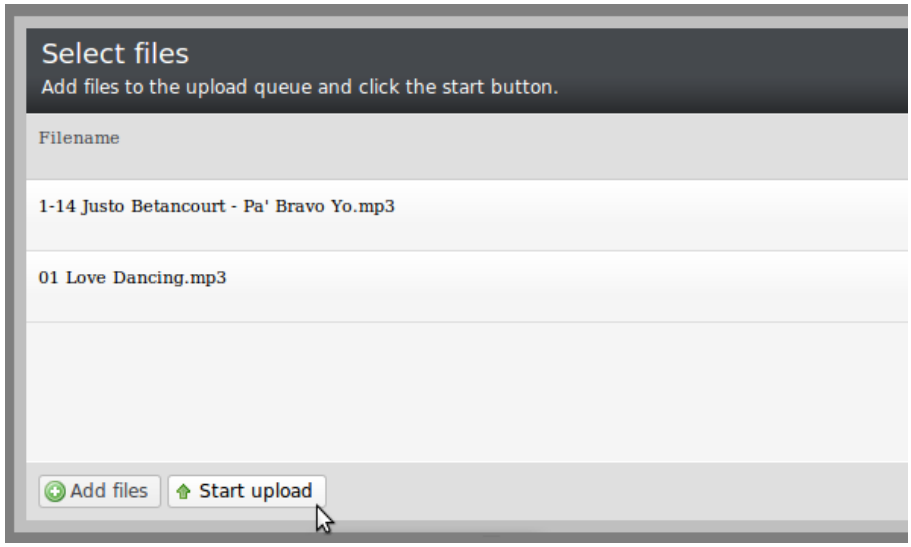
The **Add Media** page is not visible to *Guest* users.



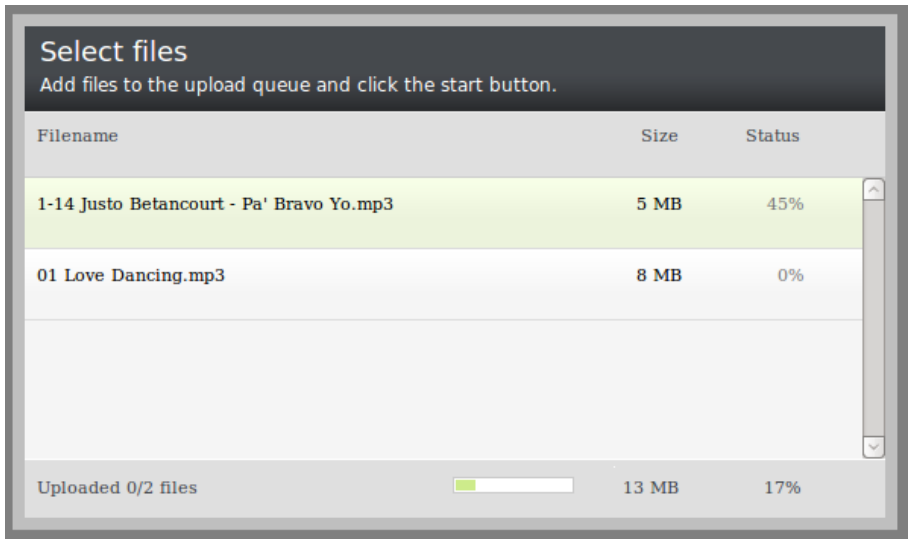
If your web browser does not support drag and drop, you can use the **Add files** button, which has a white plus sign in a green circle icon, to open a file selection window on your computer.



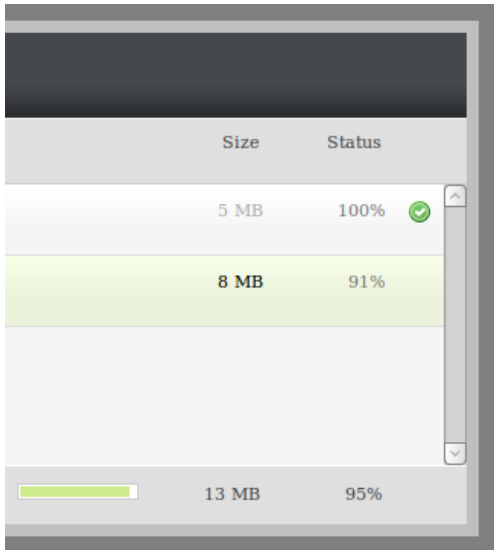
After you have added all the files that you require to the upload queue, click the **Start upload** button, which has a green arrow icon.



The row of the file currently being uploaded is highlighted in pale green. At the bottom of the upload queue, a progress bar and percentage indicates how much of the upload has taken place so far.



Once it has been uploaded successfully, each file row displays a white check mark in a green circle icon.



Your files are now imported into the Airtime storage and database, ready to be included in your broadcast playlist.

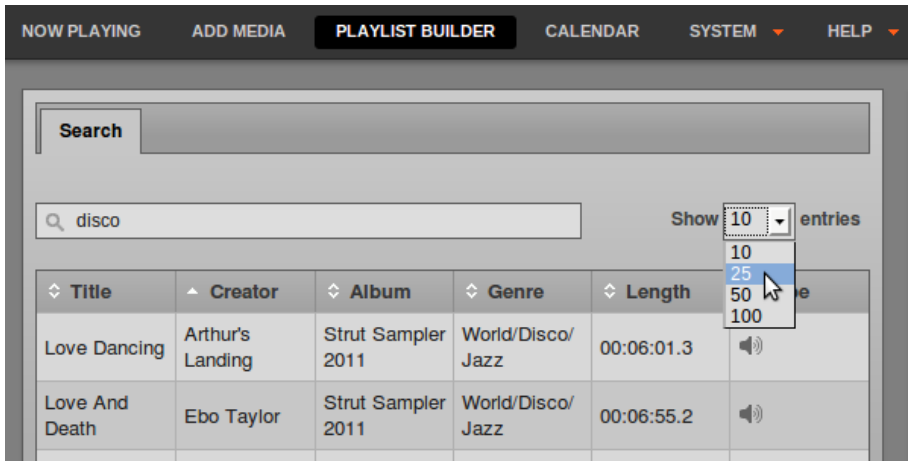
12. PLAYLIST BUILDER

This page of the Airtime interface enables you to search the media archive, sort and display the search results by the criteria that you choose, and drag and drop search results into an open playlist. You can also re-arrange the currently open playlist by dragging and dropping.

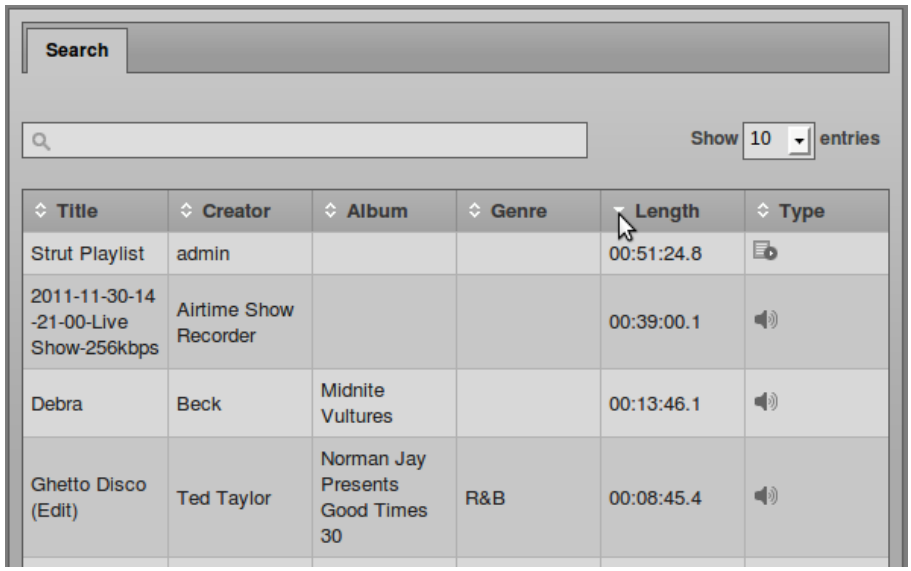
The **Playlist Builder** page is not visible to *Guest* users.

THE SEARCH TAB

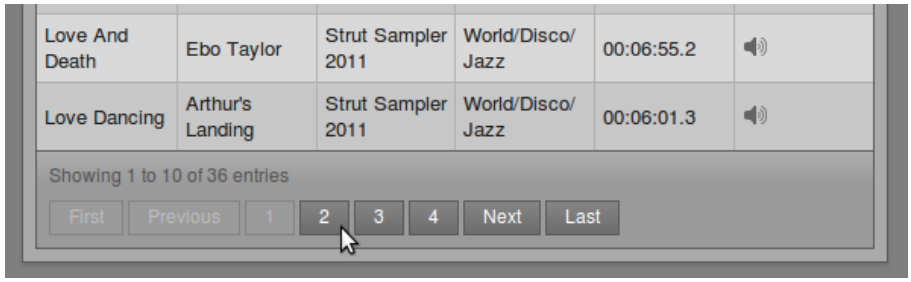
On the left hand side of the Playlist Builder page, the **Search** tab lists both the media and the playlists in the Airtime storage archive, with ten entries shown per page by default. Click on the **Show entries** drop-down menu in the top right corner to select the display of 10, 25, 50 or 100 entries per page, according to your preference.



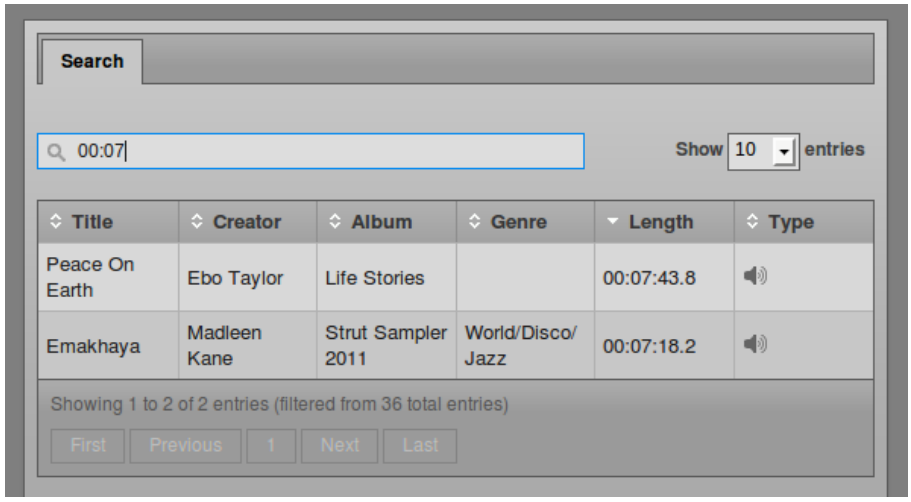
Click the column headings **Title**, **Creator**, **Album**, **Genre**, **Length** or **Type** to sort the entries. In the **Type** column, audio files are represented by a loudspeaker icon, while playlists are represented with a document icon.



At the bottom of the window, click the **First**, **Previous**, **Next**, **Last** or individual page number buttons to browse the media archive.

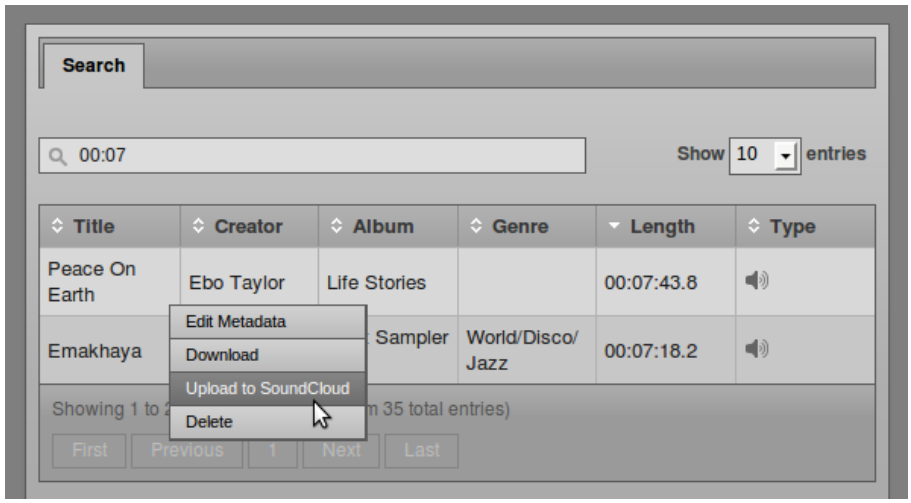


Type a search term into the box in the upper left corner, with the magnifying glass icon. You can search by **Title**, **Creator**, **Album**, **Genre**, or **Length**. For example, to search for all files and playlists between seven and eight minutes long, enter *00:07* into the search box. Like an Internet search engine, the filtered entries shown are refined as you type - there is no need to press the **Enter** key on your keyboard.

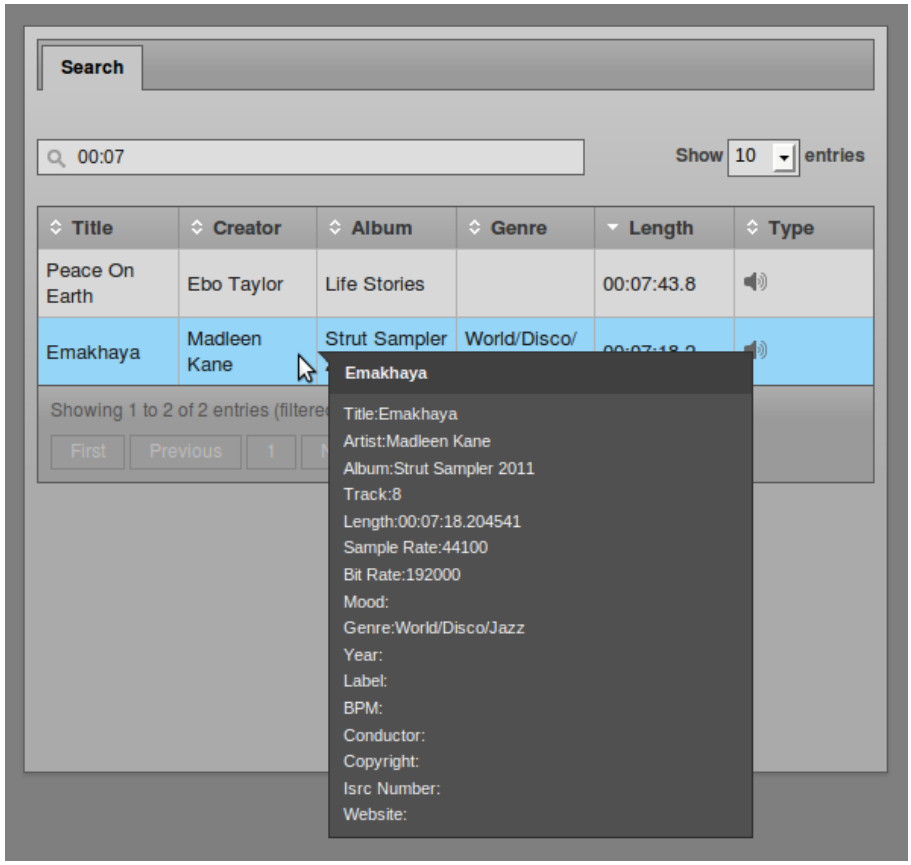


Clicking on a file row in the **Search** tab opens a pop-up menu which enables you to **Edit Metadata** for that file, **Download** it to your local computer or mobile device for audition or editing, or **Delete** it from the Airtime storage server. The **Delete** option should be used with caution, because this action cannot be undone. For this reason, only *admin* users are allowed to delete files.

If you have checked the **Enable SoundCloud Upload** box on the Preferences page, there will be an additional option on the pop-up menu, **Upload to SoundCloud**. As mentioned previously, you should only upload audio files to SoundCloud with the permission of the copyright holder.

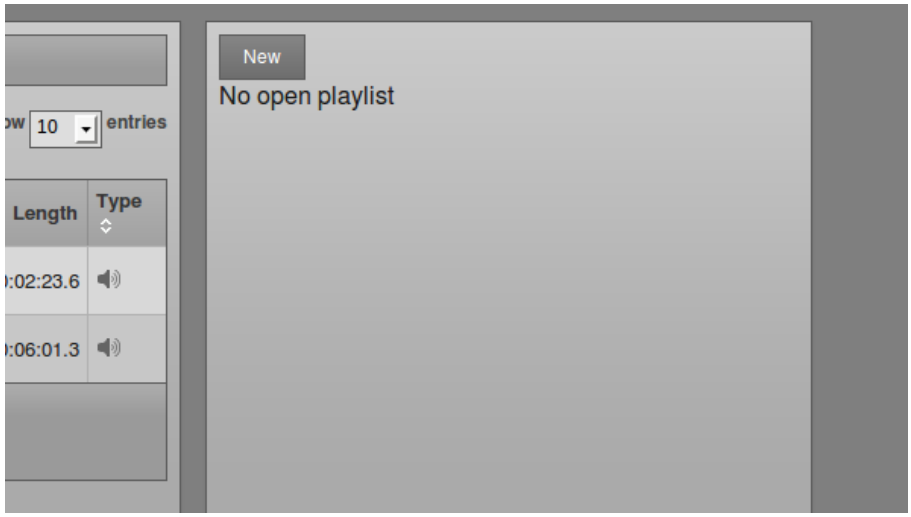


Right-clicking on a file row in the search results will display the metadata for that file in a pop-up window.

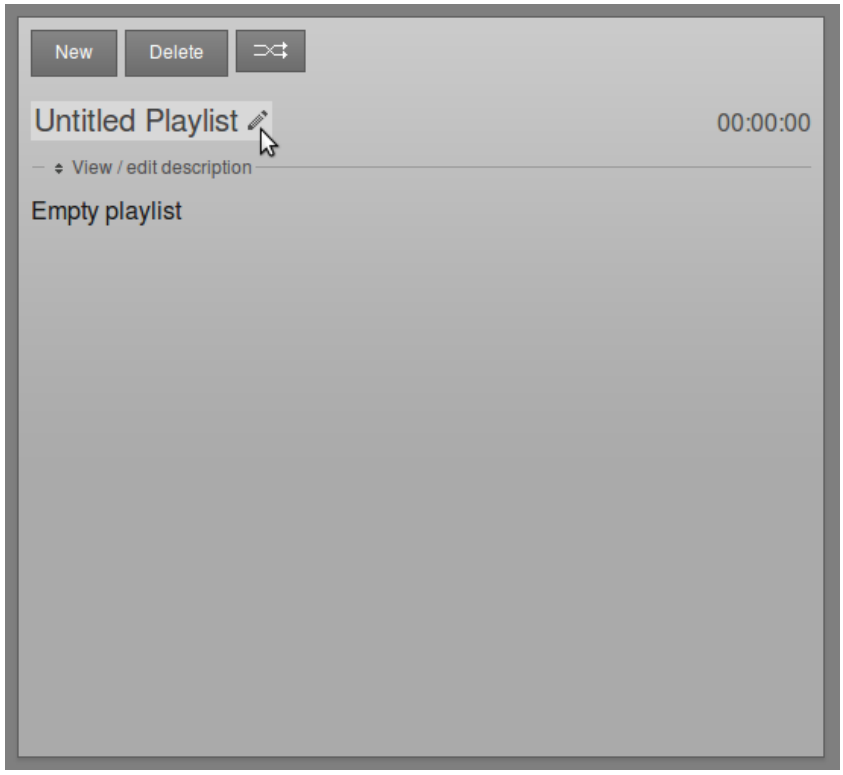


CREATING A NEW PLAYLIST

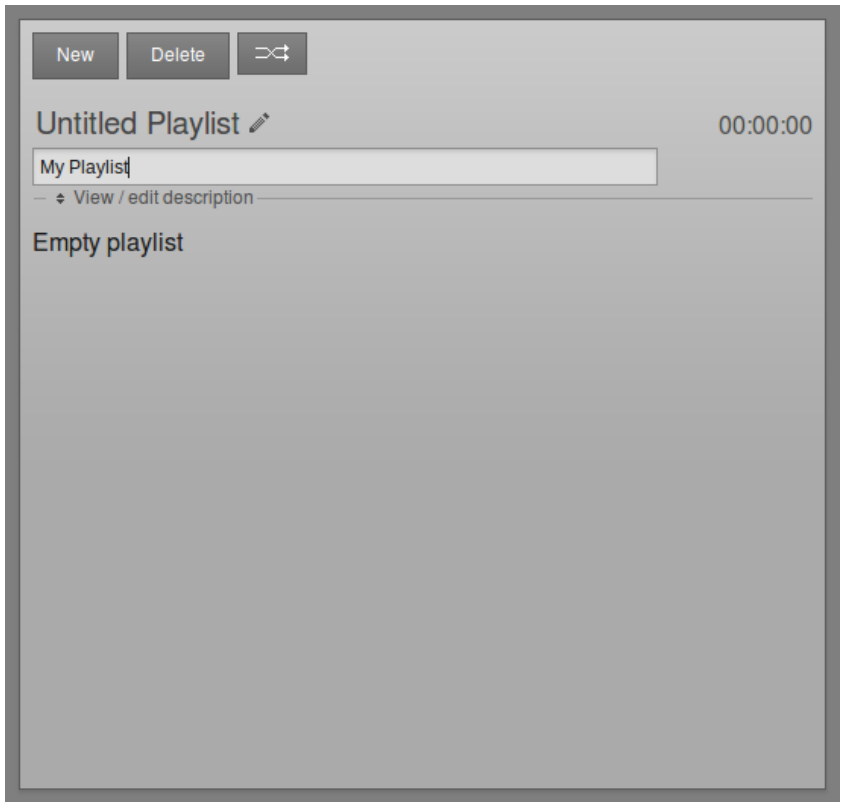
Once you have found the media that you require using the **Search** tab, you can create a new playlist on the right hand side of the **Playlist Builder** page. Click the **New** button to begin.



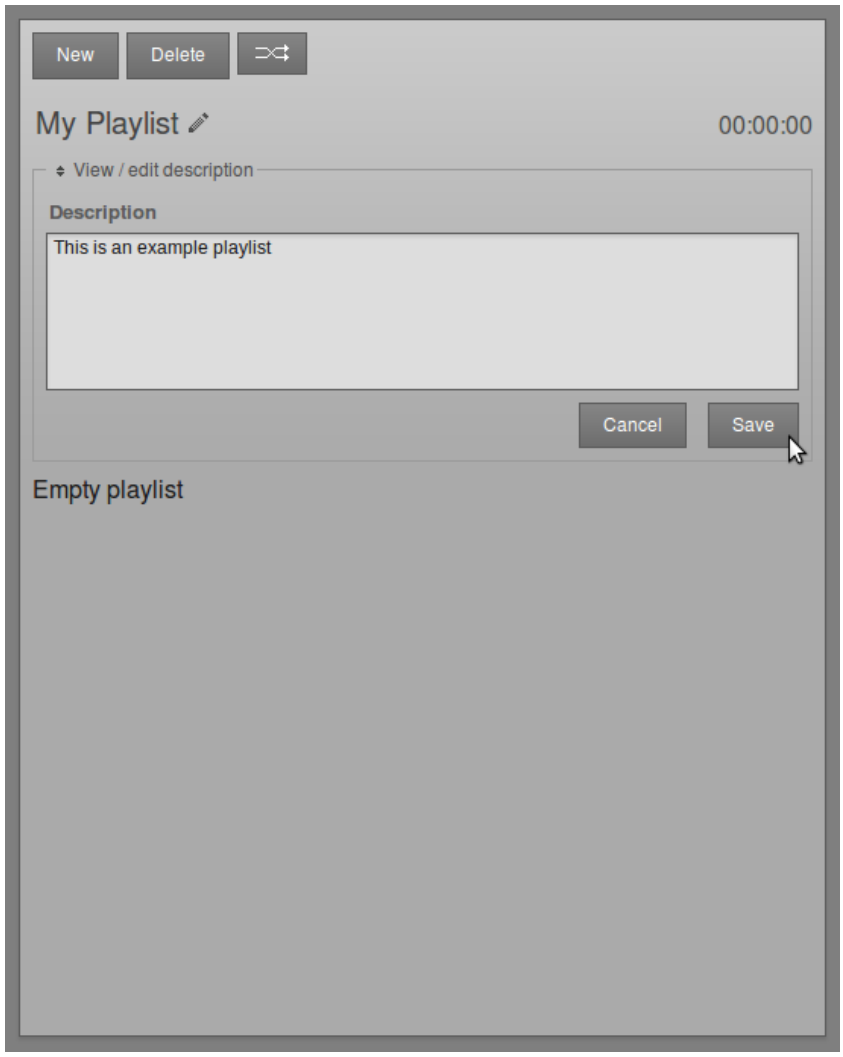
At first, the new playlist will be shown as *Untitled Playlist*. Click on the pencil icon to the right to give the playlist a name.



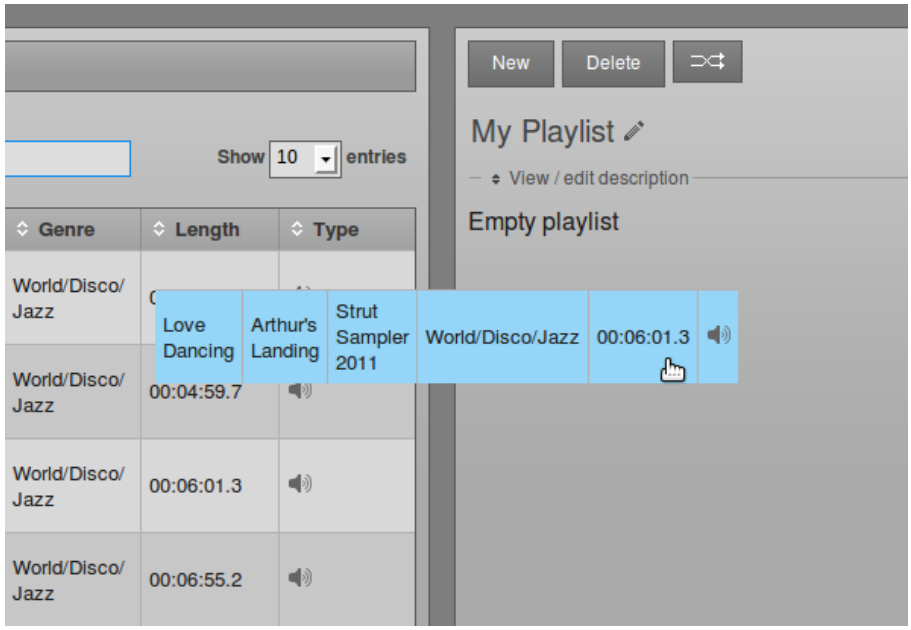
Enter the name you have chosen, then press the Return key on your keyboard to save the new name. You can edit the name of the playlist later by clicking on the pencil icon again.



Click the link **View / edit description** to expand a box where you can enter a **Description** for the playlist, then click the **Save** button. Setting good quality metadata here will help you find the playlist using the **Search** tab later, so you should be as descriptive as possible.

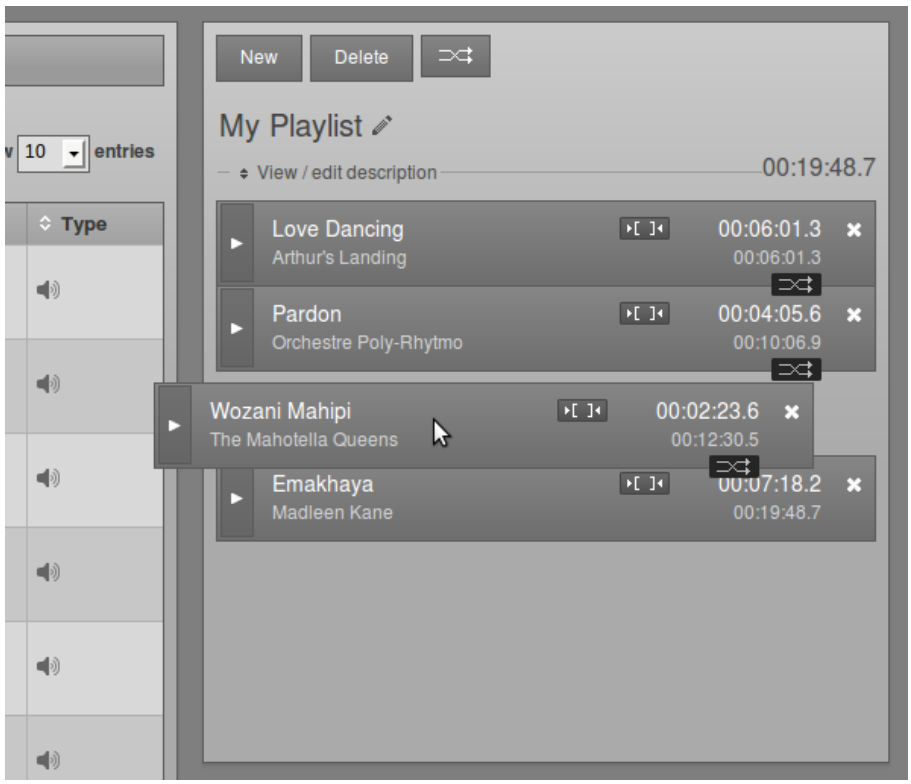


Next, drag and drop files from the Search tab on the left into the new playlist on the right.

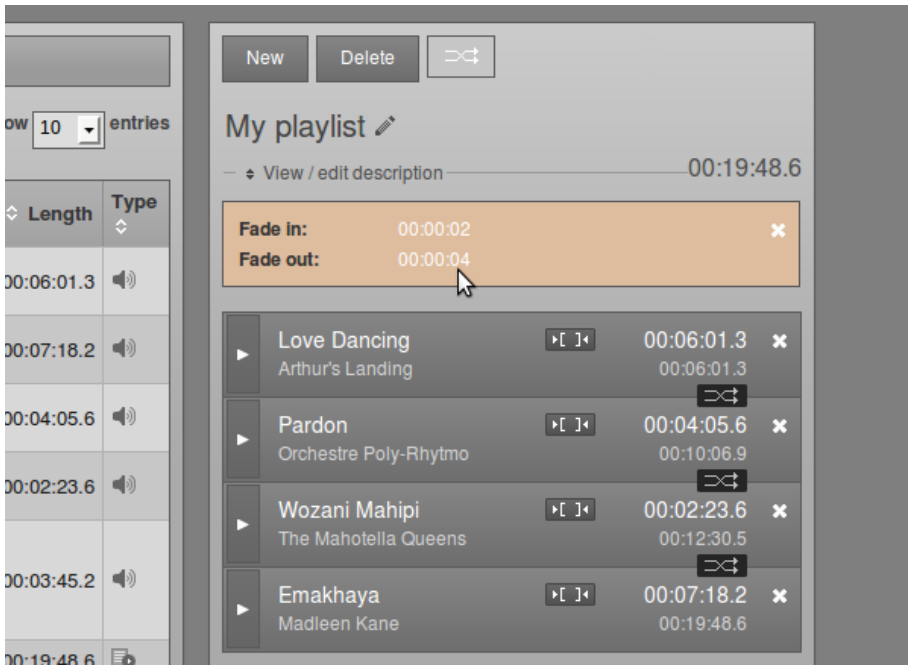


After dragging files into the new playlist, the total playlist time is displayed in the top right corner. The duration of an individual file is shown in each row of the playlist in a white font, and beneath this figure the time since the beginning of the playlist is displayed in a smaller light gray font.

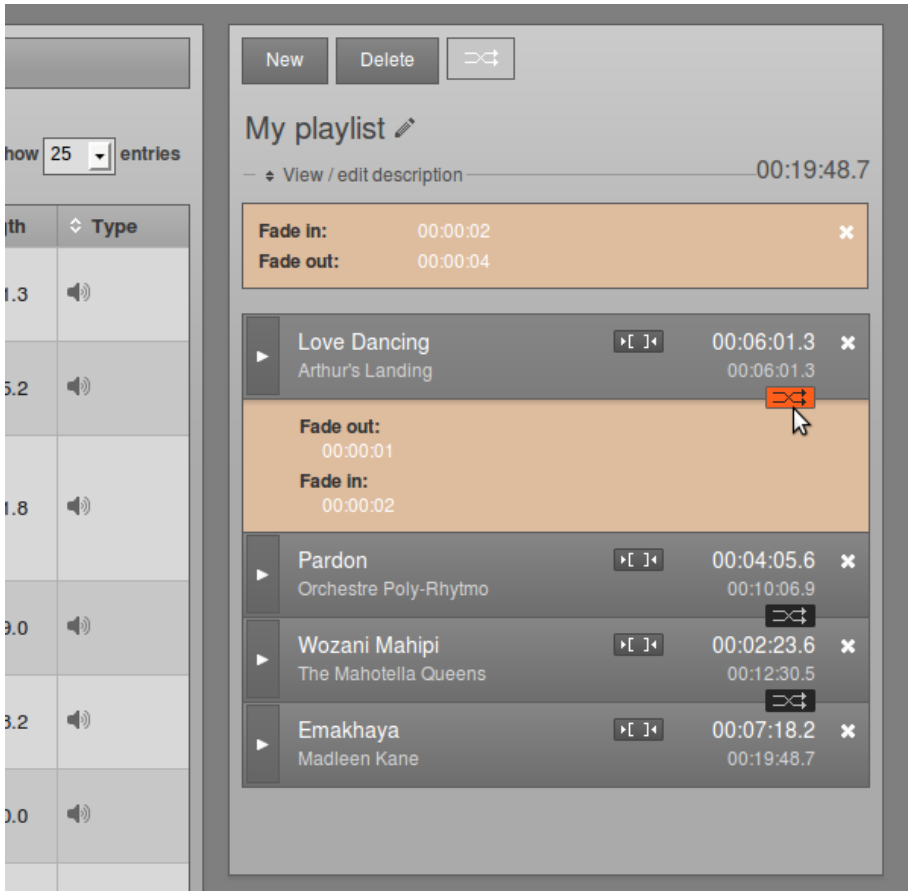
To audition a playlist file in your web browser, click the white triangle **play/pause** button on the left side of each row. Click the small white **x** icon on the right hand side of each row to remove a file from the playlist. You can also drag and drop files to re-order them.



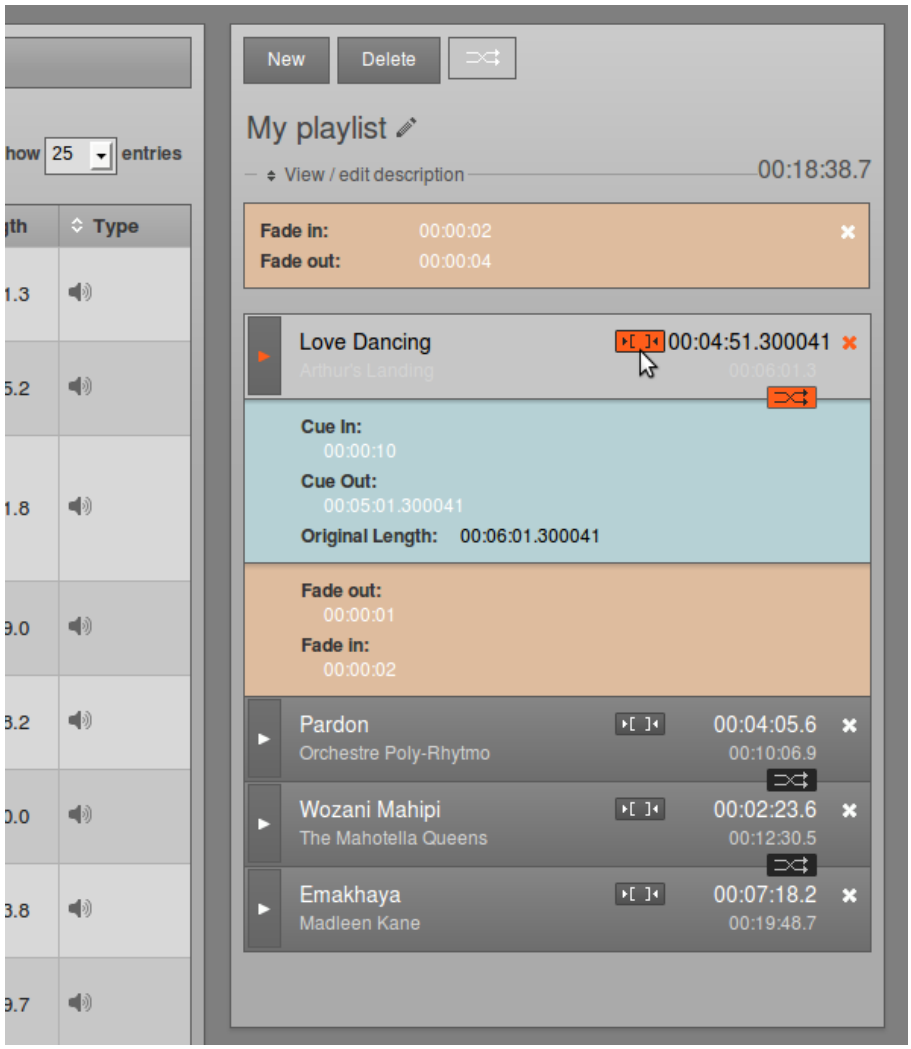
Click the playlist **Fade** button (two horizontal white arrows crossing in a gray rectangle), to the right of the **New** and **Delete** buttons, to open a beige bar in which you can set the **Fade in** and **Fade out** times for this playlist.



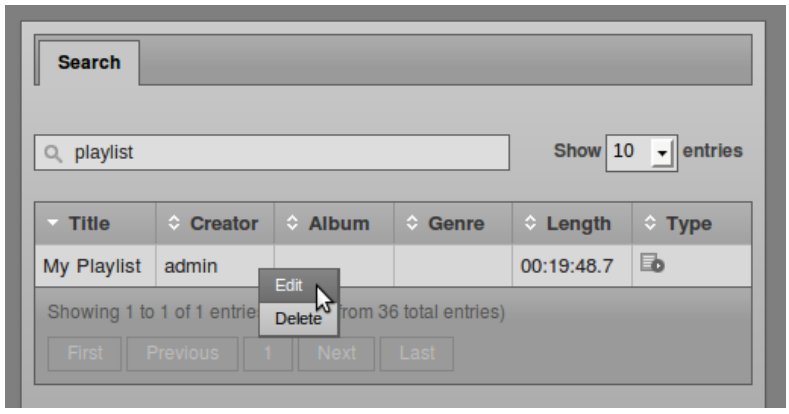
Click any one of the smaller **Fade** buttons between file rows to open another beige bar which enables you to set fade in and fade out times between two adjacent files in the playlist. The fade buttons for adjacent files change to an orange background when you click them.



Each file in the playlist also has a button with two square brackets, which enables you to set cue in and cue out times for that particular file. Like the fade button, it changes to an orange background when you click it. The length of the file in the playlist is updated automatically, but the **Original Length** of the file is also displayed for your reference.

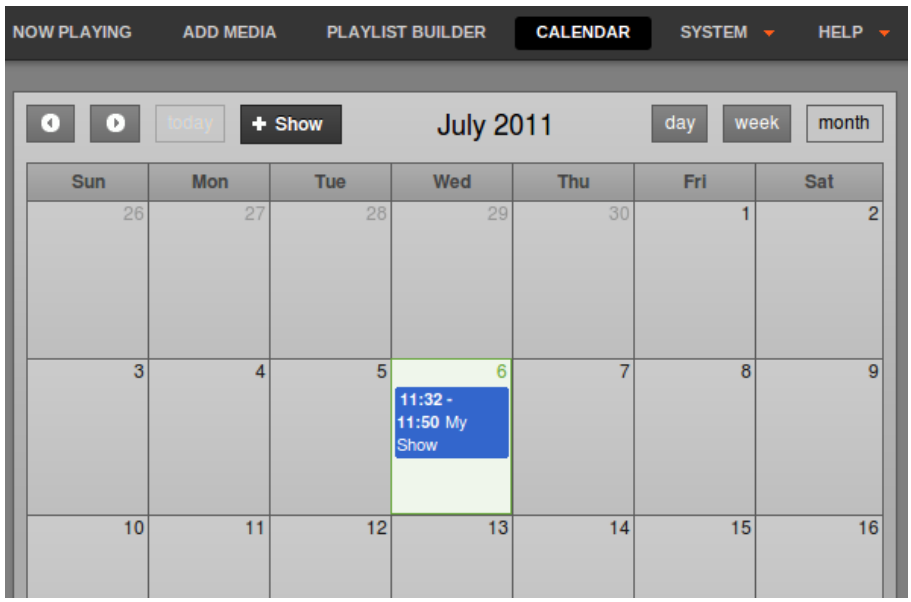


When your playlist is complete, click the **New** button in the top left corner to create another playlist, or browse to another page of the Airtime interface. If you want to edit the playlist contents or the playlist metadata later, you can find it by **Title**, **Creator** or **Length** using the **Search** tab of the **Playlist Builder** page. Click on the playlist in the search results list, and select **Edit** from the pop-up context menu. A playlist that is already open for editing will show the option **Close** in the context menu instead. You can also **Delete** a playlist from this context menu.

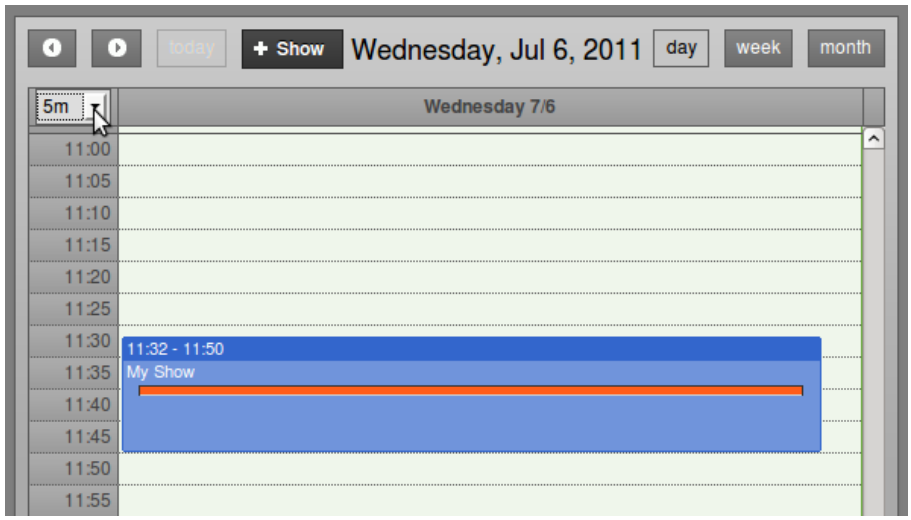

Search
playlist
Show 10 entries
Title
Creator
Album
Genre
Length
Type
My Playlist
admin
00:19:48.7
Edit
Delete
Showing 1 to 1 of 1 entrie
rom 36 total entries)
First
Previous
1
Next
Last

13. CALENDAR

The Calendar page of the Airtime administration interface has three views: **Day**, **Week** and **Month**, which can be switched using the gray buttons in the top right corner. By default, the **Month** view is shown, with today's date highlighted by a pale green background.



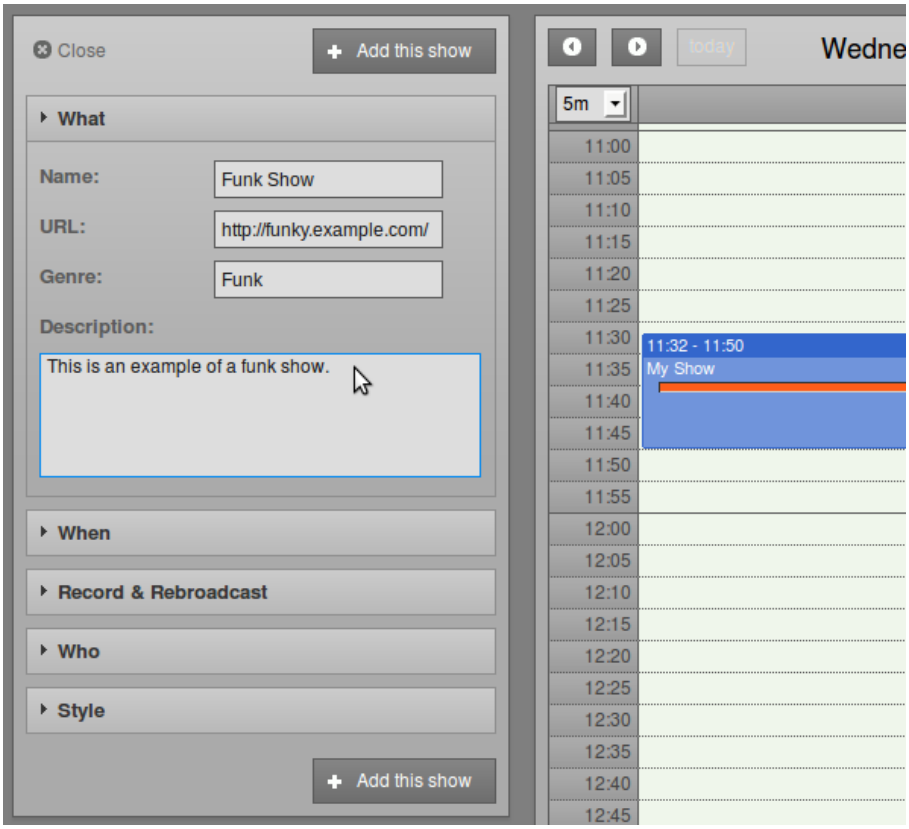
In the top left corner of the page, you can go back or forward through the **Calendar** by clicking on the buttons which have a small gray triangle in a white circle. Click the **Today** button to jump to today's date in the current view. In the **Day** or **Week** views, there is also a drop-down menu which allows you to set the resolution displayed for the calendar, ranging from one minute per row to sixty minutes per row.



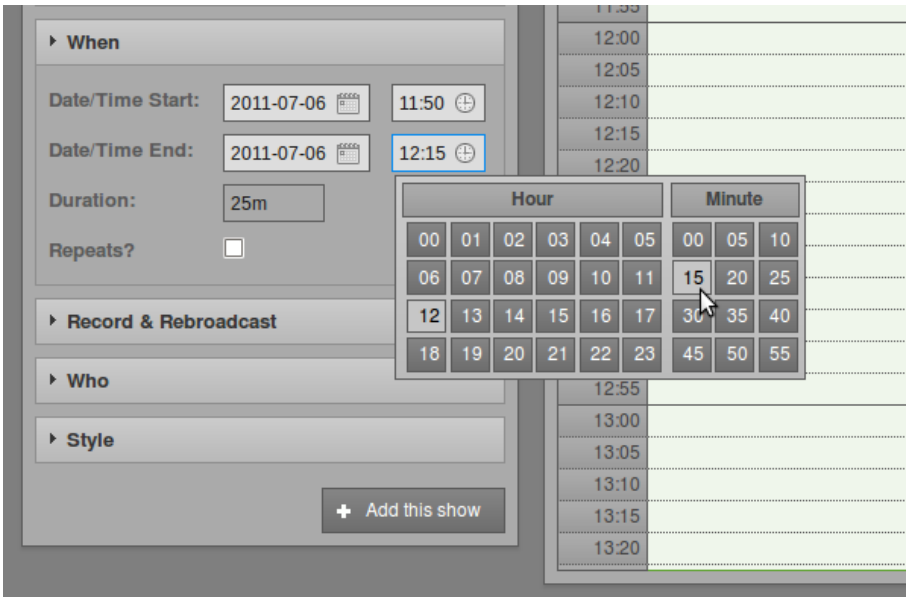
ADDING A SHOW

Only Admins and Program Managers can use this feature. To add a new show to the Calendar, click the **+ Show** button in the top left corner of the page, or click on any empty row in the Calendar itself. Either of these actions opens the **Add this show** box, which has five sections, arranged vertically: **What**, **When**, **Record & Rebroadcast**, **Who**, and **Style**. Click the small black triangle to the left of the section name if you wish to minimize or maximize it.

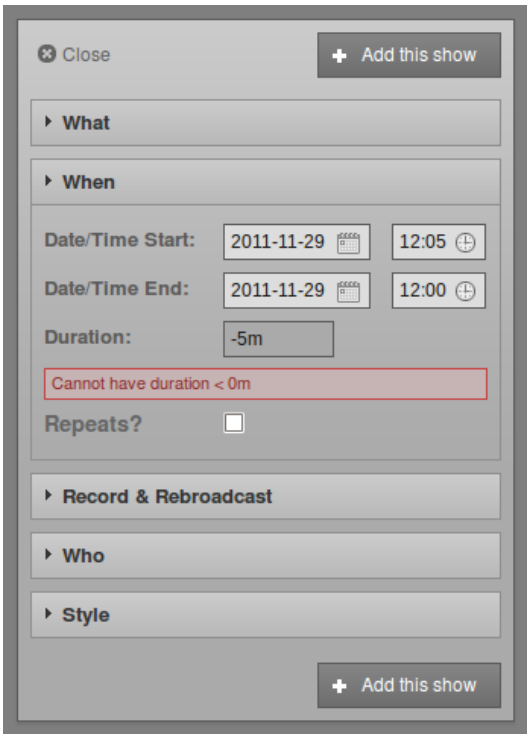
In the **What** box, enter the name, public website URL, genre and description for the show that you are creating.



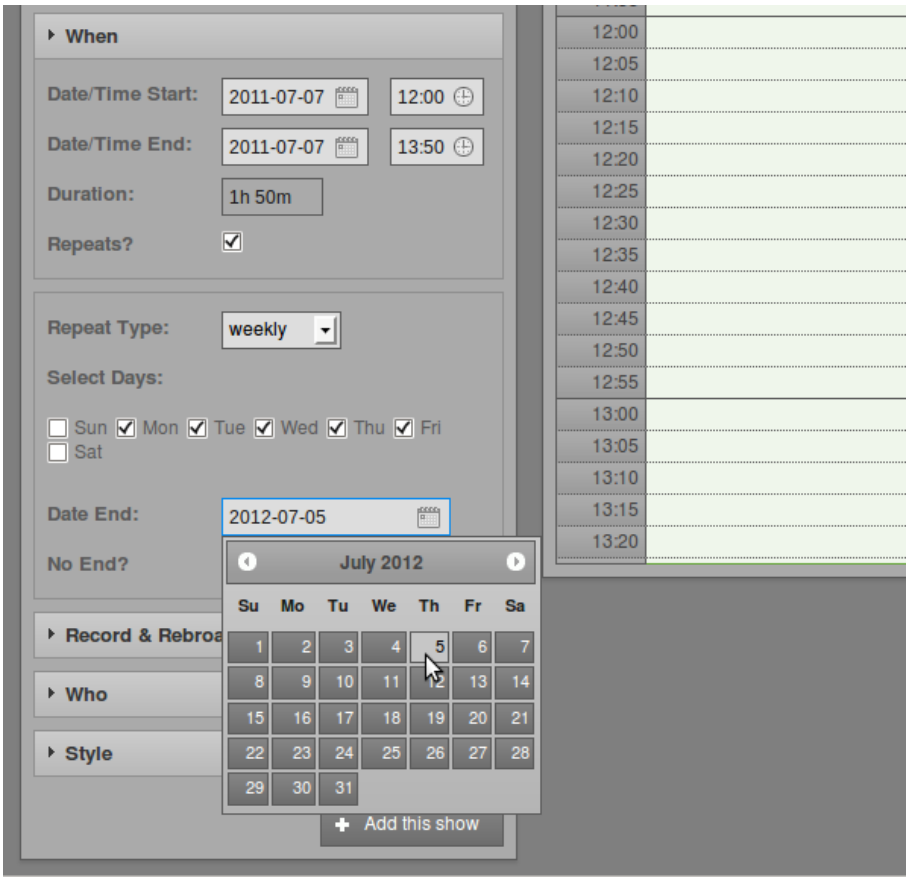
Next, in the **When** section, if you clicked on a date in the **Calendar** this should already be entered in the **Date/Time Start** field. To set another date for the show, click on the date in the **Date/Time Start** field and select the alternative date that you require for the show from the small pop-up calendar which will appear. Click on the adjacent time field to set the start time for the show, with the pop-up **Hour** and **Minute** box. The minute values in the pop-up time boxes are rounded to the nearest five minutes. You can adjust the times manually by clicking into the fields and typing. Repeat the process to set the **Date/Time End** fields. The **Duration** of the show will be displayed automatically, based on the start and end times you have set.



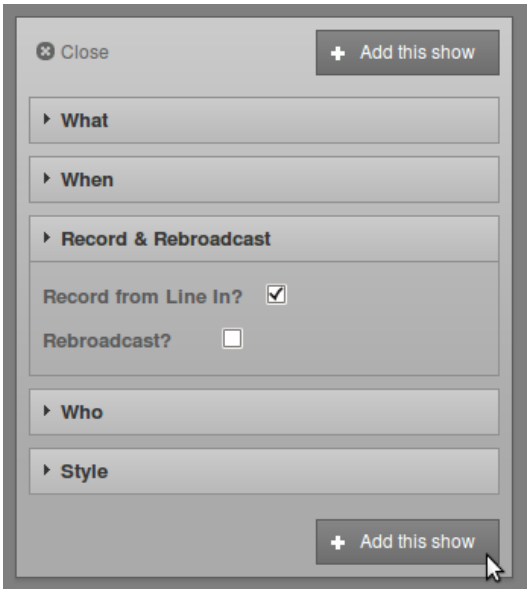
Airtime will only allow you to enter valid times for shows. If you attempt to schedule show times which would be impossible, Airtime will display a warning message.



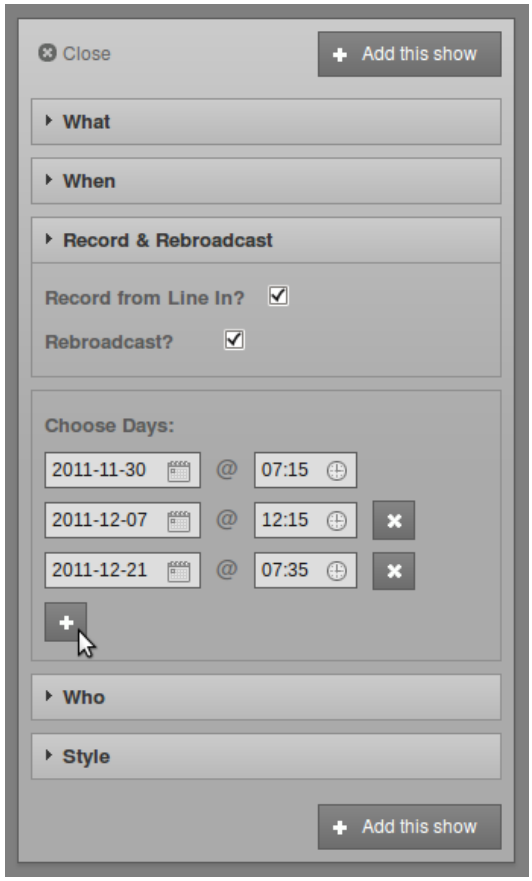
To schedule a regular show, check the **Repeats?** box and select either **Weekly**, **Bi-weekly** or **Monthly** from the **Repeat Type** drop-down menu that will appear. Then check the boxes for the days of the week that you want to schedule the regular show on. Set the **Date End** for the regular show to finish, or check the **No End** box to schedule the show indefinitely.



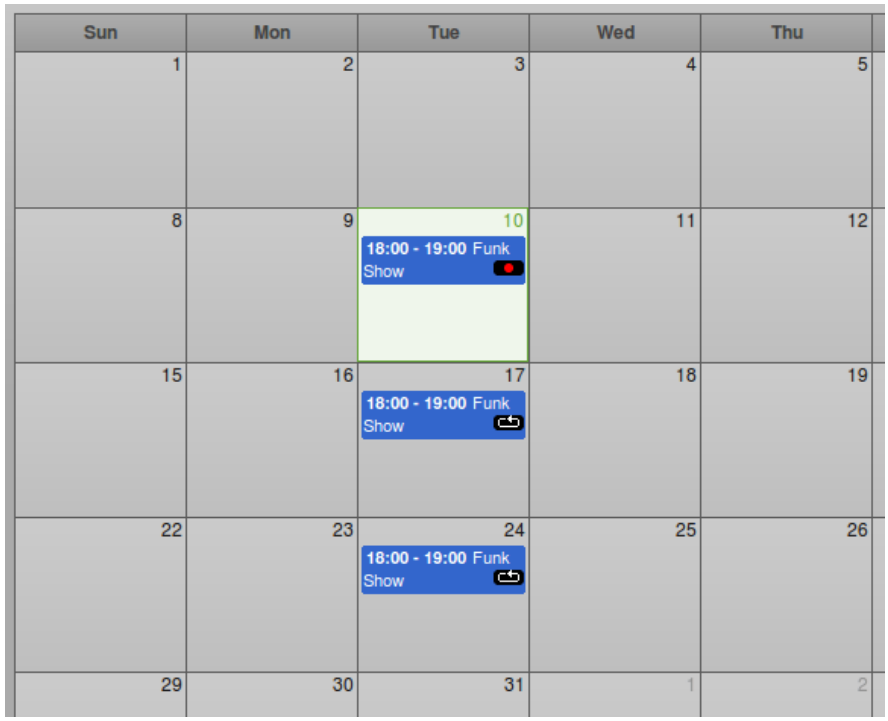
In the **Record & Rebroadcast** section, the **Record from Line In?** box enables automatic recording of the soundcard line input, if your Airtime server has one, at the time of the show. Shows set for live recording cannot also contain playlists. The audio format for live recordings is Ogg Vorbis, and the recording files are saved in the storage directory that was set in the **Manage Media Folders** page on the **System** menu.



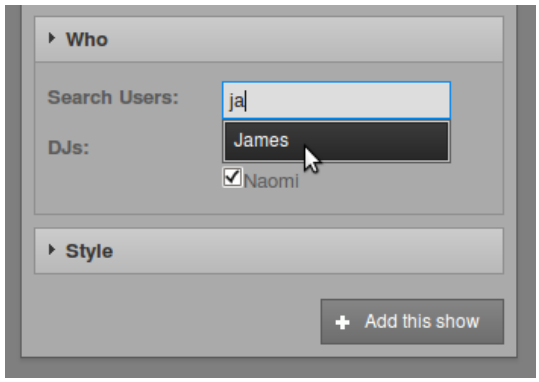
If you wish the recording to be played out at a later time, check the **Rebroadcast?** box, and then select up to ten date and time slots in the **Choose Days** box.



Shows set for recording have a small red dot icon in the calendar, while rebroadcast shows have a white loop icon.



In the **Who** section, type the first few letters of the name of the show's DJ (presenter) in the **Search Users** field to select a name from the database, or check one of the **DJs** boxes in the vertical list below. This association of a DJ name with a particular show enables that DJ to add playout media to the show, so it is important to get the DJ's name right.

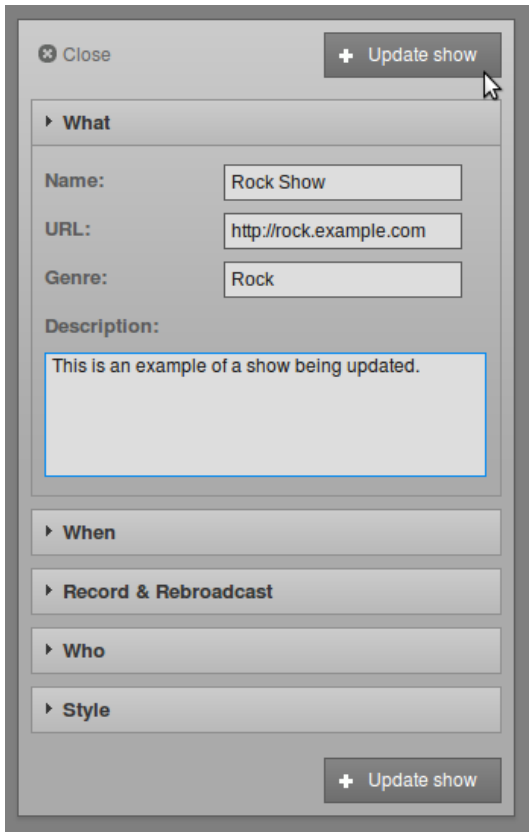


Finally, select a **Background Colour** and **Text Colour** from the **Style** section, so that the new show can be easily identified in the **Calendar**. Click the colored circle icon in the lower right corner to close this pop-up window.

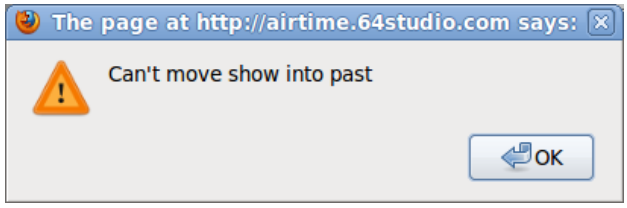
Click the **Add this show** button at the top or bottom of the box. The new show will now be displayed in the **Calendar**, with a regular slot if you have chosen to schedule one.

EDITING A SHOW

Show configuration and metadata can be changed at any time until broadcast of that show commences. Click on the show in the Calendar, and select **Edit Show** from the pop-up context menu. This opens the **Update Show** box, which is almost exactly the same as the **Add this Show** box. Click the **Update show** button at the top or bottom of the box when you are done.



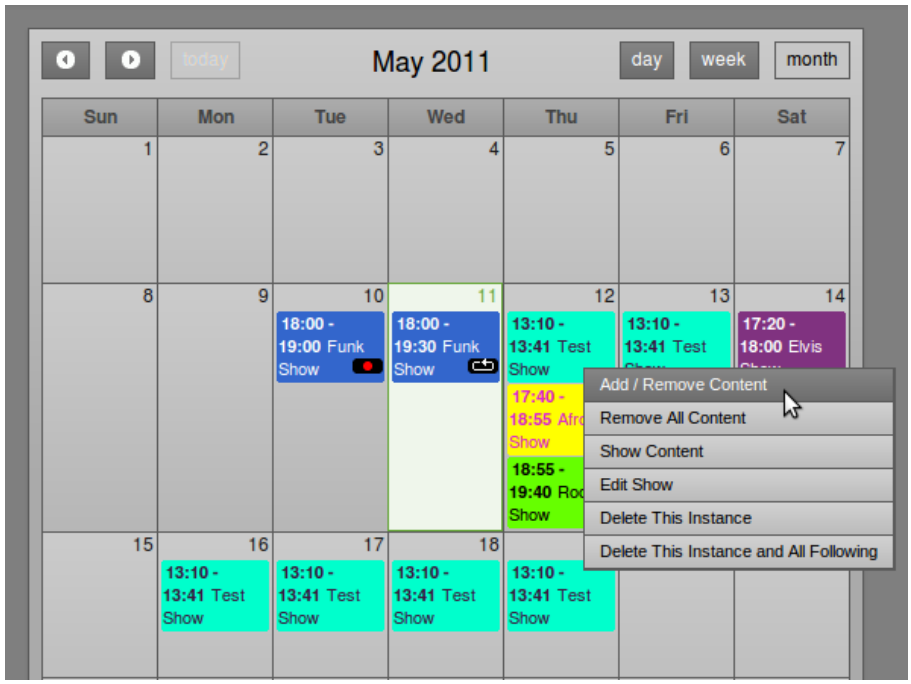
Alternatively, individual shows can be clicked on and dragged to new days and times in the calendar. However, Airtime will not allow you to drag a future show into the past, or drag a show onto a day where this would cause show times to overlap. You cannot drag and drop instances of a repeated show, either.



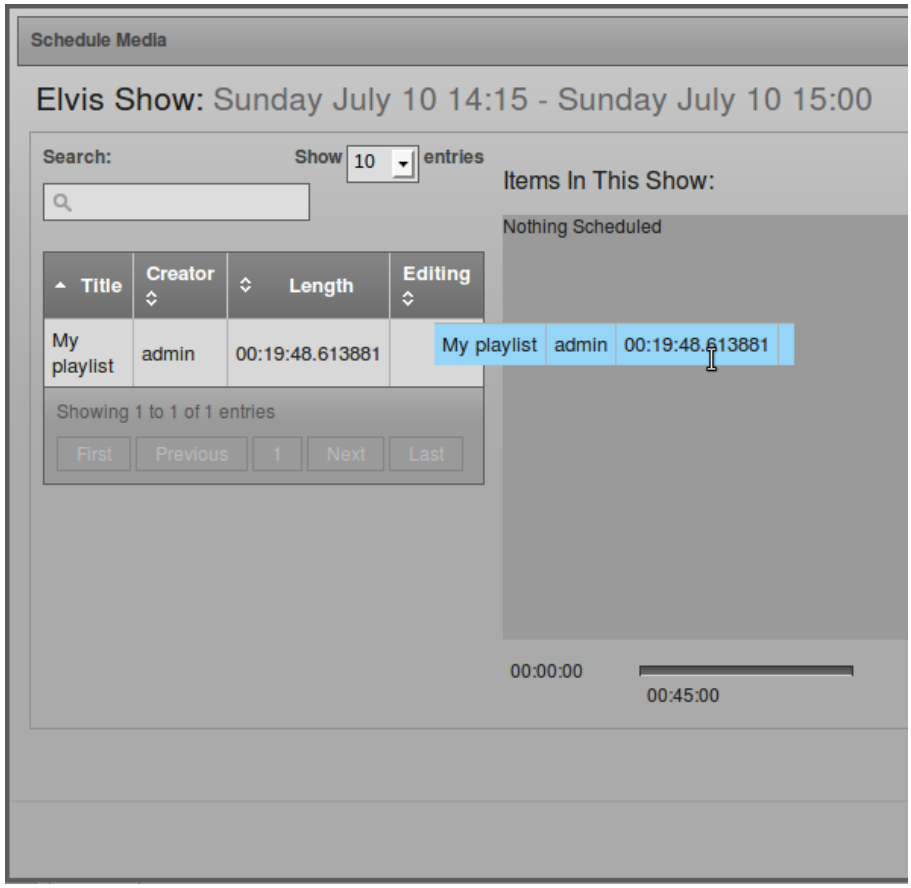
In the **Day** and **Week** views, show length can be adjusted by clicking on the lower edge of the show box, and dragging the edge of the box upwards or downwards. The new show length is calculated automatically. Airtime will not allow you to drag a show end time beyond the start of the next show.

ADDING CONTENT TO A SHOW

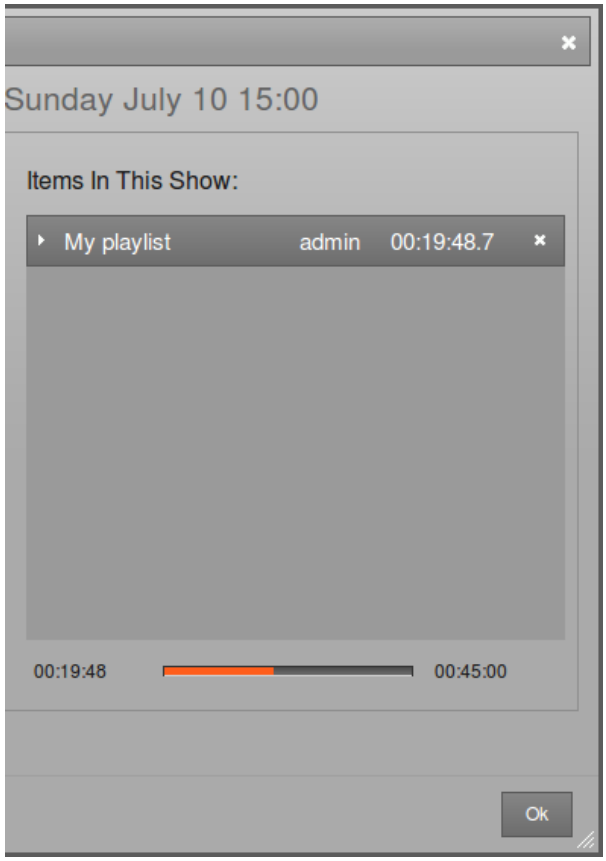
To add content to a show, click on it in the Calendar, and select **Add/Remove Content** from the pop-up context menu.



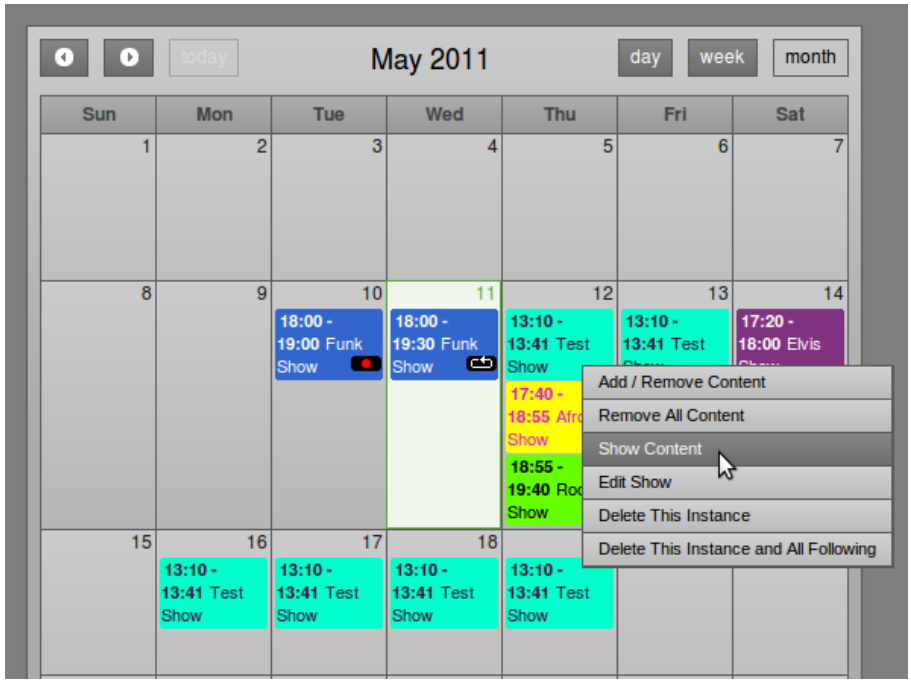
This action opens the **Schedule Media** pop-up window. Just like when using the **Playlist Builder** to find files, you can search for playlists and drag them into the **Items In This Show** box. If a member of the station staff is editing the playlist in question at the time, the name of that user will be shown in the **Editing** box.



The orange progress bar underneath the **Items in this Show** box updates automatically to show how much time remains from the allocated show duration. If you add too much media relative to the length of time allocated to the show, Airtime will display a warning, and playout will be cut when the show ends. Click the small white triangle on the left of each row to see details of the playlist, or click the white **x** icon on the right side to remove it from the show.



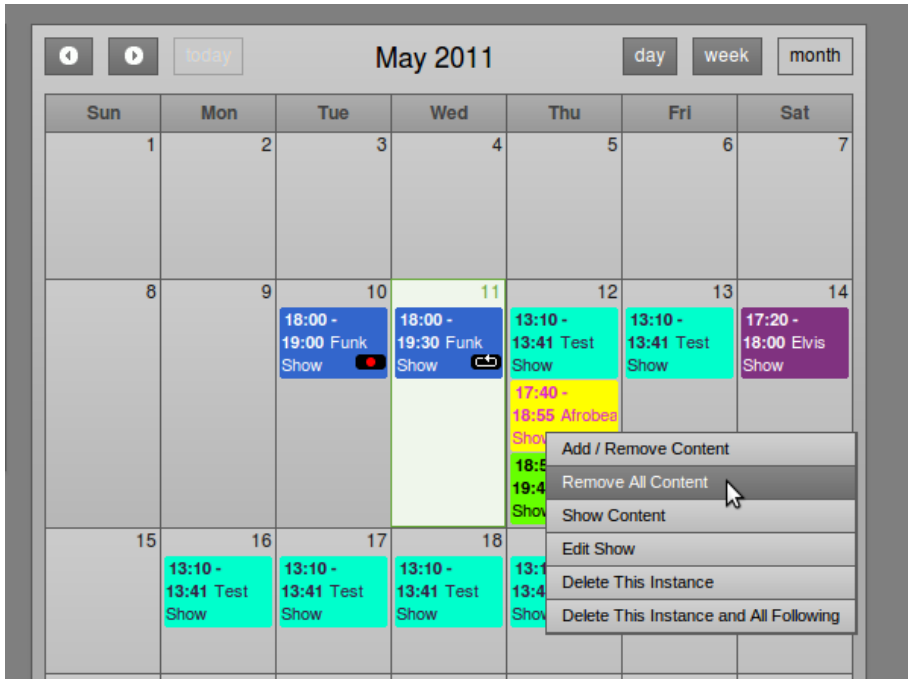
When your show has all the required content, click the **OK** button in the bottom right corner to close the window. You should make sure that you click **OK** before the show's scheduled start time. Back in the **Calendar**, click on the show and select **Show content** from the pop-up context menu to view a list of content now included in the show.



Click on any row in the **Show Contents** window to download the individual file for audition, or further editing. Once you're done, click the **OK** button in the bottom right corner, or the white **x** icon in the top right corner, to close the window.

REMOVING CONTENT FROM A SHOW

To remove an individual playlist from a show, click on the show in the **Calendar**, and select **Add/Remove Content** from the pop-up menu. In the **Schedule Media** window which opens, use the white **x** icon to remove the playlist from the show content. To remove all playlists from a show, click on the show in the **Calendar**, and select **Remove All Content** from the pop-up menu.



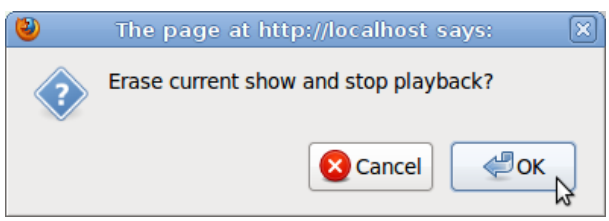
DELETING A FORTHCOMING SHOW

To delete one instance of a forthcoming show, click on the show in the **Calendar**, and select **Delete This Instance** from the pop-up menu. If you wish to delete all future instances of this particular show, select **Delete This Instance and All Following** from the pop-up menu.

You cannot delete or remove content from shows that have already played out. These shows have only one option on the pop-up menu, which is **Show Content**.

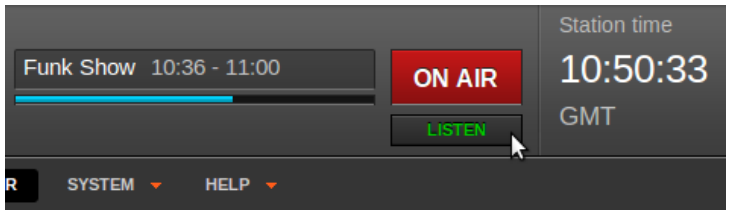
CANCELING PLAYOUT

If you wish to cancel playout of a playlist based or pre-recorded show while it is running, click on the show in the **Calendar** and select **Cancel Current Show** from the pop-up menu. Airtime will ask you if you are sure about this action, as it cannot be undone.

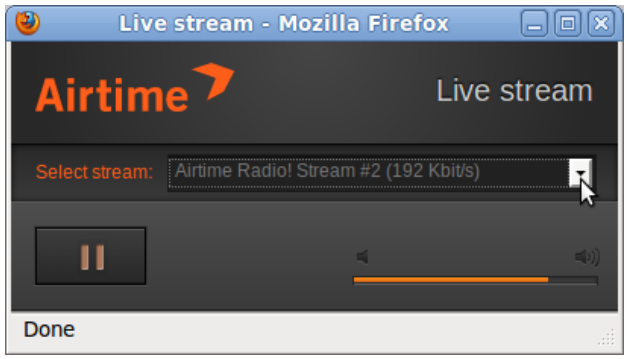


14. LISTEN

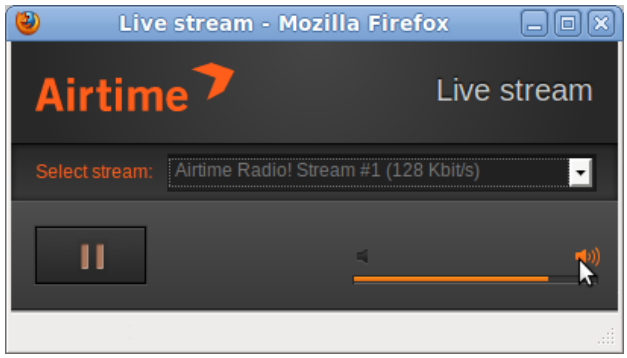
In the Master Panel, beneath the **ON AIR** indicator, you will find the **LISTEN** button.



This button opens a pop-up **Live stream** window, which enables you to monitor the streams that have been configured previously in the **Stream Settings** page on the **System** menu. In the **Live stream** window, a drop-down menu enables you to switch between the streams which are currently available.



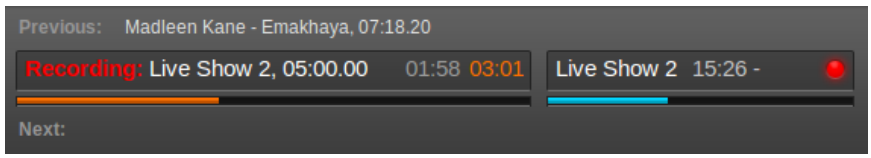
Beneath the drop-down menu for stream selection is a rectangular pause/play button on the left, and an orange volume control bar on the right. This volume control only adjusts the output level of the pop-up **Live Stream** window, not the output level of the Airtime server itself. Click on the left side speaker icon to mute the output, or click on the right side speaker icon to raise output to maximum level.



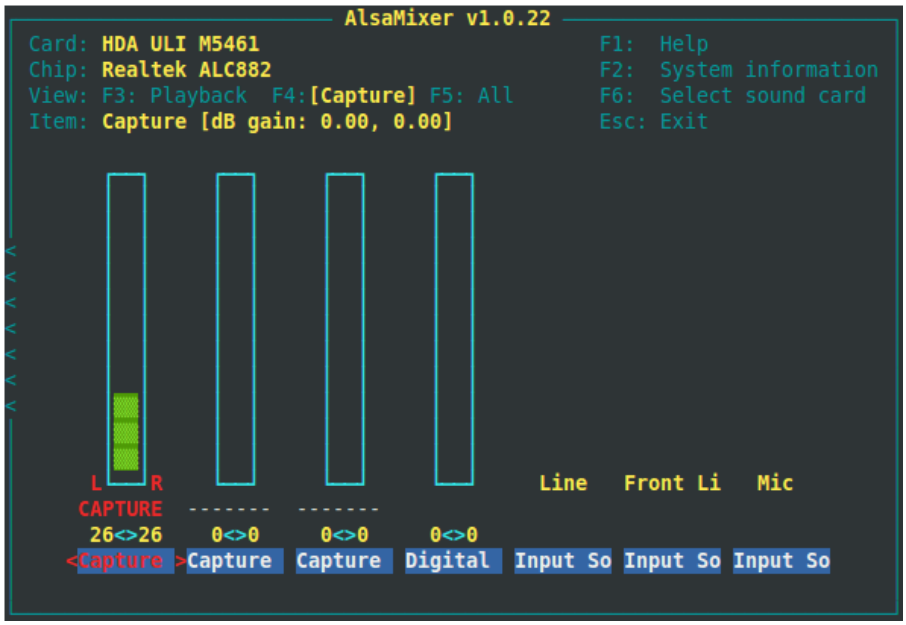
To adjust output level in between muted and maximum, click on the corresponding place in the orange bar. When you have finished monitoring the streams, you can close the pop-up window in the normal way, depending on the browser you are using. In Firefox, you can close the window by clicking the **X** button in the top right corner. This action will not shut down the output from the Airtime server, only the stream monitoring on your desktop computer.

15. RECORDING

Live show recording from the line input of the server soundcard, if one is fitted, can be enabled in the **Add Show** box of Airtime's **Calendar** (see the chapter *Calendar* in this book). During a recording, a red light is shown in the **Show** box of the Master Panel.



Before a broadcast show scheduled for recording begins, you should check that the level of recording is sufficient to ensure a good signal to noise ratio, but not so high a level that clipping (distortion) occurs in the recorded file. You can perform this check using the command **alsamixer** on the Airtime server. This command opens a soundcard mixer application in the server console.



Press the **F4** key on your keyboard to set capture levels. Some experimentation may be required to find the correct control on the mixer, using the **Left** and **Right** arrow keys on your keyboard. Levels are set with the **Up** and **Down** arrows, and *Capture* is toggled with the **Space** bar. *Capture dB gain* should be set to *0.00, 0.00* initially.

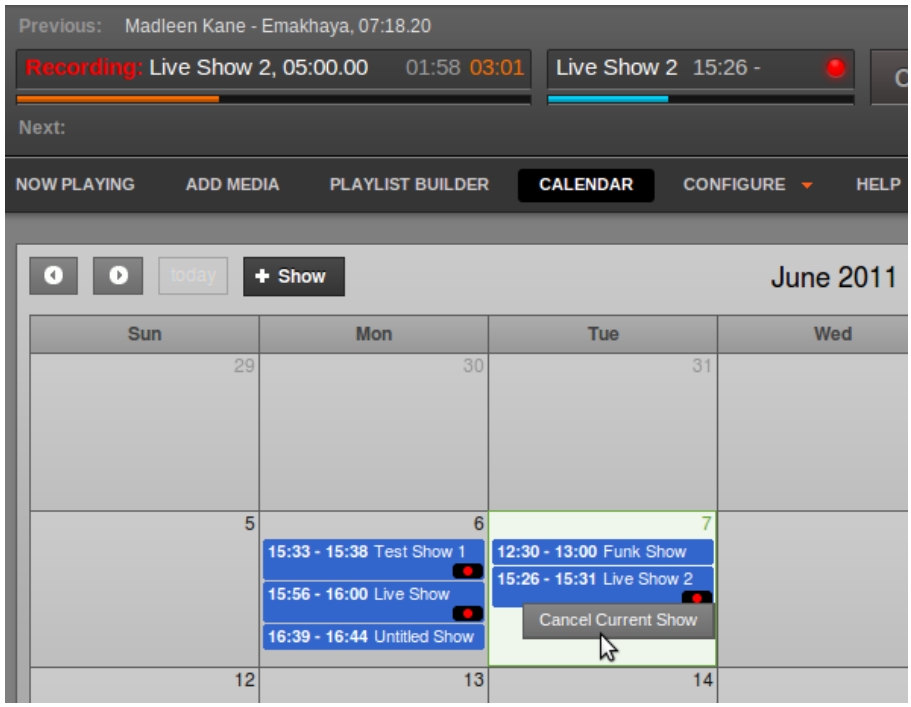
If you hear nothing at all in the recording, you may need to set the value of *Input Source* to *Line*, using the **Up** or **Down** arrows. Depending on the particular soundcard and sockets you are using, you may have to enable other inputs, such as *Digital* or *S/PDIF*. If a test recording is too quiet, try raising the line output level of your broadcast mixer before increasing gain on the soundcard, in order to achieve the optimal *gain structure*.

USING RECORDINGS

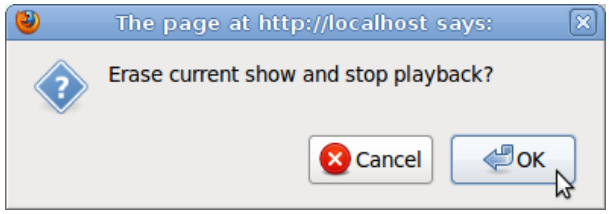
After the recording has finished, you can find the recorded file in the **Playlist Builder**, by searching for **Recorder**. The file will be labelled with a **Title** containing a date and time stamp, as well as the name of the show.

CANCELING A RECORDING

If you wish to cancel the recording of a live show and delete the recorded file from the storage server, click on the show in the **Calendar** and select **Cancel Current Show** from the pop-up menu.

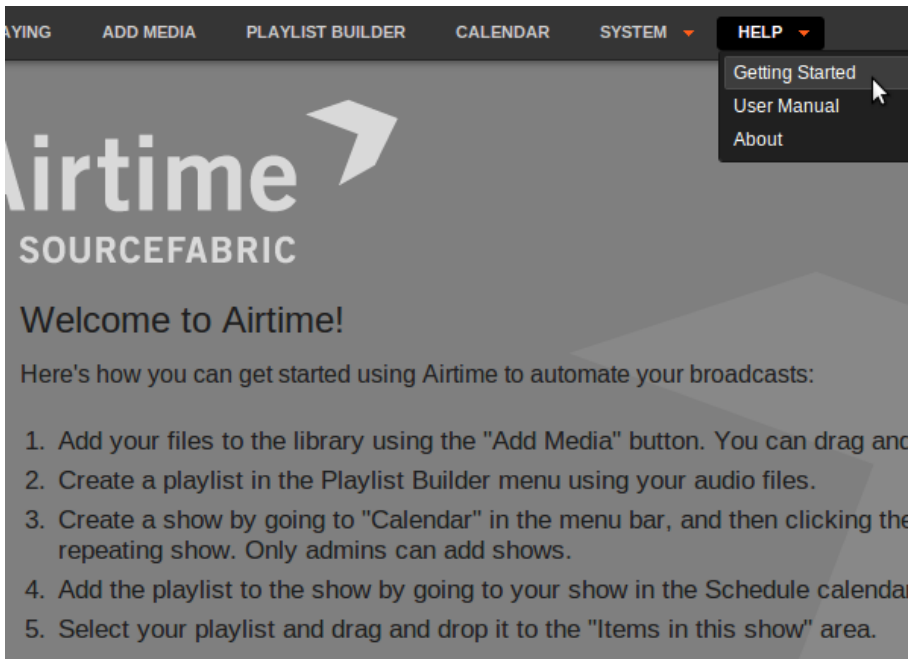


Airtime will ask you if you are sure about this action, as it cannot be undone.



16. HELP

The first entry on Airtime's **Help** menu offers a **Getting Started** guide for new users. Further down, there is also a link to the online version of this **User Manual** and an **About** page, which displays version and licensing information.



You can visit the Airtime online support forum, and sign up for the mailing list, at http://forum.sourcefabric.org/categories/airtime-support

This forum is mirrored by the mailing list, so posts on the forum appear on the mailing list and vice versa. You can therefore also post a message there by emailing airtime-support@lists.sourcefabric.org

To subscribe to forum updates via mail, please register or login to the forum by clicking the appropriate link. Then click the 'Subscribe' button at the top of each forum page.

BUG REPORTING

Airtime needs your input to improve. If you think you've found a bug, please visit http://dev.sourcefabric.org/ and sign in, using the same login and password that you registered for the Airtime forum. Create a bug report by selecting **Create Issue**, then **Airtime**, and then **Bug**. That way, the Airtime team can keep track of your problem and notify you when it has been fixed. You can also suggest improvements and new features for Airtime on that site.

CONTACT

Finally, when all other avenues have been exhausted, email us directly at contact@sourcefabric.org and we'll try to help!

OTHER HELP

The UNESCO publication *Community Radio - A user's guide to the technology*: http://cr-userguide.web4all.in/CommunityRadioUserGuide.pdf features a very comprehensive guide to setting up a community radio station. This guide is aimed at people thinking about setting up a radio station in India, but includes lots of practical advice that would be useful in any country.

INSTALLATION

17. PREPARING THE SERVER

The following instructions assume that you have root access (*sudo* on Ubuntu) to a GNU/Linux server, and are familiar with basic command line tasks. Experienced system administrators may prefer to skip to the *Expert install* chapter in the appendix of this book after preparing the server as shown in the steps below.

The recommended Airtime server platform is Ubuntu 10.04 'Lucid Lynx' LTS. Ubuntu 11.04 'Natty Narwhal', 11.10 'Oneiric Ocelot', and Debian 6.0 'squeeze' are also supported options. The server should have at least a 1GHz processor and 512MB of RAM, preferably 1GB RAM or more. If you are using a desktop environment and web browser directly on the server you should install at least 2GB RAM, to avoid swapping to disk.

The Airtime installation does not use much disk space, but you should allow plenty of storage capacity for the media archive. A hot-swap RAID array is recommended for the media archive, in case of disk failure. You should also consider a UPS or other battery powered system to offer some protection against short-term power failures.

The Airtime web administration interface is intended to work with any browser, on any desktop or mobile platform with a minimum display size of 1024x768 pixels. The recommended web browser is **Mozilla Firefox 3.6** (or a later version). **Google Chrome 8** (or later) and **Apple Safari 4** (or later) are also supported.

SOUNDCARDS

If your Airtime machine will only be used to stream directly to an Icecast or SHOUTcast streaming media server, you do not require a soundcard to be installed on the Airtime server side. This option is suitable for Airtime installations at your ISP's datacenter, remote from any transmitter. However, you will not be able to take advantage of Airtime's live show recording feature.

If you intend that your Airtime server will have a direct audio output to a broadcast transmitter or a separate stream encoder, your server machine must have a soundcard supported by an ALSA driver. Almost all standard soundcards have ALSA drivers built into the Linux kernel, which do not need to be installed separately. If in doubt about driver support for your soundcard, check the ALSA soundcard matrix at: http://www.alsa-project.org/main/index.php/Matrix:Main

USB AUDIO DEVICE INDEX

Some server motherboards do not have a default ALSA device, because a USB soundcard is prevented from getting index number 0 by the GNU/Linux distribution's configuration. This setting may be in a file such as */etc/modprobe.d/alsa-base.conf* in Ubuntu, which can be edited with **nano**:

```
$ sudo nano /etc/modprobe.d/alsa-base.conf
```

Comment out the lines beginning *options snd-usb-* to fix the problem:

```
# Prevent abnormal drivers from grabbing index 0

# options snd-usb-audio index=-2
# options snd-usb-us122l index=-2
# options snd-usb-usx2y index=-2
# options snd-usb-caiaq index=-2
```

Save the file with **Ctrl+O** and close **nano** with **Ctrl+X**. Then remove and re-insert the cable connecting the USB audio device. The command aplay -l should now confirm that the USB audio device has index 0:

```
$ aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: MobilePre [MobilePre], device 0: USB Audio [USB Audio]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

When using a USB soundcard with ALSA, some how-to documents advocate setting the *nrpacks=1* option, but this is not recommended for Airtime because it can increase CPU load significantly.

INTEL HDA MIXER LABELS

If you have an **Intel HDA** soundcard, as built in to many contemporary motherboards, you may discover that the recording controls in **alsamixer** have incorrect labels. This problem can make it difficult to adjust mixer levels except by trial and error. To fix these labels, you can pass a specific *model=* parameter to the *snd-hda-intel* module of the Linux kernel. On Debian or Ubuntu GNU/Linux, you can do this by first finding the model you have, with the command:

```
$ cat /proc/asound/card0/codec* | grep Codec
```

The server should respond with a line such as:

```
Codec: Realtek ALC882
```

Referring to the appendix *HD Audio Models* in this book, find the matching codec and model. In this example, the codec is *ALC882* and the motherboard has six analog jacks and two S/PDIF sockets, so the model is *6stack-dig*.

```
ALC882/883/885/888/889
======================
  3stack-dig 3-jack with SPDIF I/O
  6stack-dig 6-jack digital with SPDIF I/O
```

Edit the file */etc/modprobe.d/alsa-base.conf* with **nano** as follows:

```
$ sudo nano /etc/modprobe.d/alsa-base.conf
```

Add an appropriate line for your soundcard model to the end of the file, such as:

```
# Realtek ALC882
options snd-hda-intel model=6stack-dig
```

Save the file with **Ctrl+O** and close **nano** with **Ctrl+X**. Then reboot the server. After the reboot, you should now see that the mixer controls are correctly labelled.

DISABLE DESKTOP AND OTHER SOUNDS

If you are installing Airtime on a desktop computer, make sure you disable or remove any programs that could send unintended audio to a soundcard you are using for broadcast output. This includes alert sounds which play when the computer is ready for use, or when a user logs in. On Ubuntu, these sounds are configured using *System -> Preferences -> Sound* on the main desktop menu. This configuration dialog only works when the PulseAudio sound server is installed.

You may prefer to remove all system sound files from the computer, in case they could be played unintentionally via the soundcard. For example, on Ubuntu you may wish to remove the **ubuntu-sounds** package, with the following command:

```
$ sudo apt-get purge ubuntu-sounds
```

REMOVE PULSEAUDIO, IF INSTALLED

The PulseAudio sound server is not compatible with Airtime soundcard output, but is installed by default on Ubuntu. To remove PulseAudio from an Ubuntu machine, type the following command:

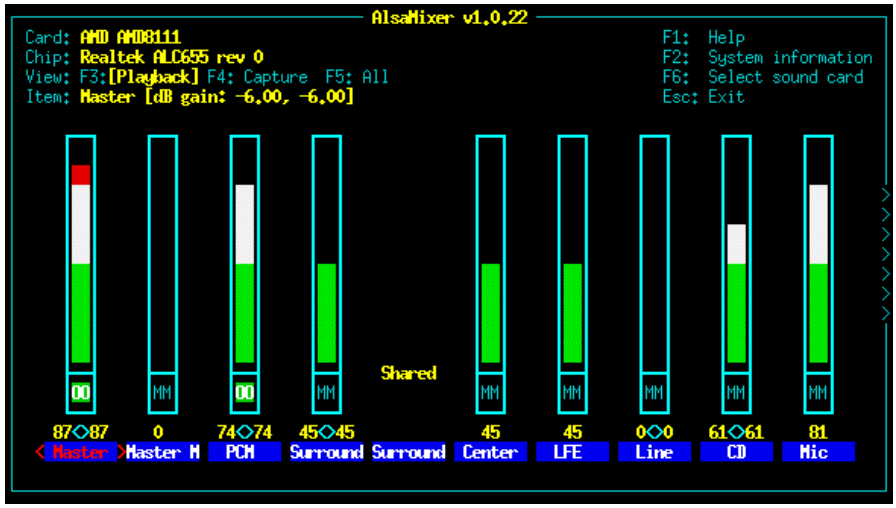
```
$ sudo apt-get purge pulseaudio
```

Removing the *pulseaudio* package on a desktop Ubuntu machine may force the removal of the *ubuntu-desktop* metapackage. However, this metapackage is only installed on the system for managing upgrades; removing it does not remove the GNOME desktop.

After removing PulseAudio, if your Airtime machine has a desktop, you can install a mixer applet which can control the ALSA soundcard driver directly, such as *gnome-alsamixer*:

```
$ sudo apt-get install gnome-alsamixer
```

On a server with a soundcard but without a desktop, you can control sound input and output levels using the command line program *alsamixer*:



This program should already be installed on an Ubuntu or Debian machine with a soundcard. If not, you can install it with the command:

```
$ sudo apt-get install alsa-utils
```

REMOVE WEBMIN, IF INSTALLED

The *webmin* control panel (http://www.webmin.com) has been known to remove Apache and PHP packages on Debian and Ubuntu systems, which can cause the Airtime package to be removed in turn. This problem is easily reversed by re-installation of the affected packages, but it has the potential to disrupt your broadcast playout from Airtime. Webmin is not likely to be installed on your server unless your system administrator has installed it manually. This is because *webmin* was removed from official Debian and Ubuntu package repositories some years ago.

RABBITMQ HOSTNAME

RabbitMQ requires a fixed and resolvable hostname (see http://www.rabbitmq.com/ec2.html#issues-hostname), which is normal for a server. For a desktop or laptop machine where the hostname changes frequently or is not resolvable, this issue may prevent RabbitMQ from starting. When using a desktop or laptop computer with a dynamic IP address, such as an address obtained from a wireless network, the *rabbitmq-server* daemon must not start up before the *NetworkManager* service.

RABBITMQ ON DEBIAN

In Debian 6.0 (Squeeze) the *rabbitmq-server* daemon does not start automatically after a reboot. This should be fixed before installing Airtime, to prevent problems at playout time. If the *rabbitmq-server* package was installed before the last reboot, you will need to run:

```
# invoke-rc.d rabbitmq-server start
```

as the root user before the installation of Airtime. If it is not already installed, run the following command as root:

```
# apt-get install rabbitmq-server
```

After a fresh installation, rabbitmq-server will start automatically, so there is no need to run the *invoke-rc.d* command mentioned above.

In either case, you should then edit lines 13 and 14 of the file */etc/init.d/rabbitmq-server* (as root) to show:

```
# Default-Start:     2 3 4 5
# Default-Stop:      0 1 6
```

and then run the command (as root):

```
# update-rc.d rabbitmq-server defaults
```

This should make sure that *rabbitmq-server* starts after the next reboot.

PROCEED TO INSTALLATION

Now your server should be ready for Airtime to be installed. Depending on your requirements, you should now proceed to either the chapter *Easy setup*, the chapter *Manual installation*, the chapter *Automated installation* or the *Expert install* appendix.

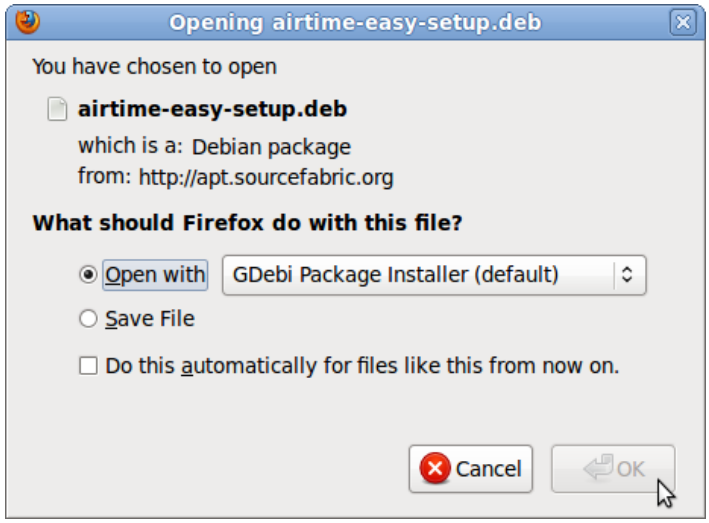
18. EASY SETUP

The *airtime-easy-setup* package sets up a typical Airtime configuration without prompting you for any settings. You can then install the latest Airtime package from the Sourcefabric repository with a few clicks.

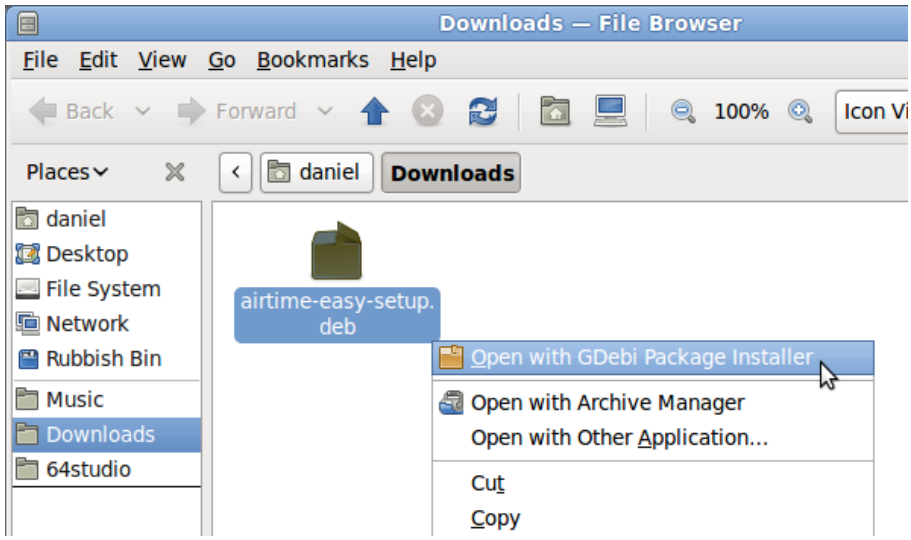
After that, the Airtime administration interface can be accessed at the default domain name of the computer (such as http://airtime.example.com). If you have not yet set a fully qualified domain name for the computer, you can use a URL such as http://ubuntu/ on the localhost for testing purposes. Whichever domain name you use, the Icecast administration interface will be at port 8000 of the same URL (such as http://airtime.example.com:8000).

You can download the *airtime-easy-setup* package from http://apt.sourcefabric.org/misc/airtime-easy-setup.deb which is a link to the latest version of the package. You should stay connected to the Internet throughout the installation, as a number of dependency packages have to be installed from online repositories.

On Ubuntu 10.04 'lucid' or Debian 6.0 'squeeze', you can run the *airtime-easy-setup* package from your browser using the program **GDebi**.



If you have chosen to save the package to your computer instead, in the desktop file manager, right-click on the *airtime-easy-setup* package and select **Open with GDebi Package Installer**:



Or for an Ubuntu lucid or Debian squeeze server without a desktop, you can use **gdebi** on the command line:

```
$ sudo apt-get update
$ sudo gdebi airtime-easy-setup.deb
```

If gdebi is not installed, you can run this command first:

```
$ sudo apt-get install gdebi
```

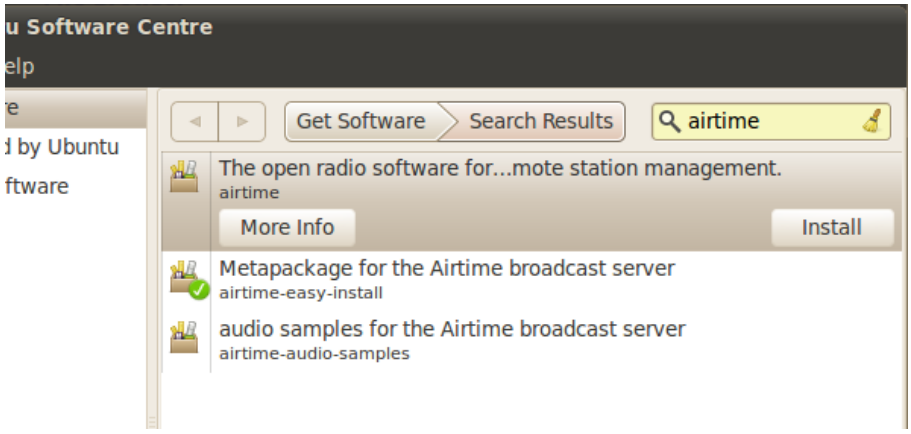
Later versions of Ubuntu have replaced GDebi with the program **software-center**. On a desktop installation, right-click on the package and select **Open with Ubuntu Software Center**. This program can also be run from the command line, for example:

```
$ software-center airtime-easy-setup.deb
```

Any of the methods above should resolve package dependencies automatically.

INSTALL AIRTIME

Once the installation of *airtime-easy-setup* has completed, you can install the latest Airtime package on a desktop system using your usual package manager, such as **Ubuntu Software Center**, or **Synaptic** on Debian.



On a server, you can use the commands:

```
$ sudo apt-get update
```

```
$ sudo apt-get install airtime
```

CONFIGURE FOR PRODUCTION

To convert a test installation into a production installation, you can run the command:

```
$ sudo dpkg-reconfigure airtime
```

The **dkpg-reconfigure** command will run through the configuration steps shown in the *Automated installation* chapter, so that you can set the correct hostnames and passwords for your production Airtime server.

19. MANUAL INSTALLATION

You do not normally need to install Airtime manually, unless you are testing a development version of the software. Versions of Airtime recommended for production use are available for download and upgrade via *secure apt*, as shown in the *Automated installation* chapter.

UPDATING PYTHON-VIRTUALENV

Airtime requires a version of **python-virtualenv** later than 1.4.8, but Ubuntu Lucid includes the older version 1.4.5 of this package. Before performing a manual installation on Lucid, you should update
python-virtualenv using the backported package available from the http://apt.sourcefabric.org/ repository. This step is not necessary when performing an automated installation, in which dependencies are resolved automatically.

FULL INSTALL

The **airtime-full-install** script has been tested on Ubuntu GNU/Linux servers and is designed to configure your server for you, using typical default settings.

1. In the server terminal or console, download Airtime from https://sourceforge.net/projects/airtime/files/ with **wget**. For example, to download version 2.0.0, you could use the command:

```
$ wget http://downloads.sourceforge.net/project/airtime/2.0.0/airtime-2.0.0.tar.gz
```

2. Unzip the downloaded file in your home directory. This action will create a subdirectory called *airtime-2.0.0*:

```
$ sudo tar -xvzf airtime-2.0.0.tar.gz -C ~/
```

3. Run the **airtime-full-install** script:

```
$ sudo ~/airtime-2.0.0/install_full/ubuntu/airtime-full-install
```

The installation script will indicate which files are being installed on your system, and the directories they are being unpacked into. Finally, it will run the **airtime-check-system** script to confirm that your server environment is set up correctly.

```
AIRTIME_STATUS_URL              = http://localhost/api/status/format/json/api_key/%%api_key%%
AIRTIME_SERVER_RESPONDING       = OK
KERNEL_VERSION                  = 2.6.32-35-generic
MACHINE_ARCHITECTURE            = x86_64
TOTAL_MEMORY_MBYTES             = 6128244
TOTAL_SWAP_MBYTES               = UNKNOWN
AIRTIME_VERSION                 = 2.0.0
OS                              = Ubuntu 10.04.3 LTS x86_64
CPU                             = Dual Core AMD Opteron(tm) Processor 170
WEB_SERVER                      = Apache/2.2.14 (Ubuntu)
PLAYOUT_ENGINE_PROCESS_ID       = 30110
PLAYOUT_ENGINE_RUNNING_SECONDS  = 15280
PLAYOUT_ENGINE_MEM_PERC         = 0.2%
PLAYOUT_ENGINE_CPU_PERC         = 0.0%
LIQUIDSOAP_PROCESS_ID           = 30113
LIQUIDSOAP_RUNNING_SECONDS      = 15280
LIQUIDSOAP_MEM_PERC             = 0.3%
LIQUIDSOAP_CPU_PERC             = 13.9%
MEDIA_MONITOR_PROCESS_ID        = 8374
MEDIA_MONITOR_RUNNING_SECONDS   = 21317
MEDIA_MONITOR_MEM_PERC          = 0.6%
MEDIA_MONITOR_CPU_PERC          = 0.0%
SHOW_RECORDER_PROCESS_ID        = 8400
SHOW_RECORDER_RUNNING_SECONDS   = 21317
```

```
SHOW_RECORDER_MEM_PERC          = 0.2%
SHOW_RECORDER_CPU_PERC         = 0.0%
RABBITMQ_PROCESS_ID            = 1588
RABBITMQ_RUNNING_SECONDS       = 21315
RABBITMQ_MEM_PERC              = 0.3%
RABBITMQ_CPU_PERC              = 0.0%

-- Your installation of Airtime looks OK!

********************** Install Complete ***********************
```

You are now ready to proceed to the *Configuration* chapter.

MINIMAL INSTALL

The alternative **airtime-install** script does not attempt to configure your server, an option which you may find more suitable if you have special requirements.

1. In the server terminal or console, install the list of dependencies. For example, on Ubuntu 10.04 (Lucid Lynx) LTS you could enter the command:

```
$ sudo apt-get install apache2 curl ecasound gzip icecast2 lame \
libao-ocaml libapache2-mod-php5 libcamomile-ocaml-data libesd0 \
libmad-ocaml libmp3lame-dev libportaudio2 libpulse0 libsamplerate0 \
libsoundtouch-ocaml libtaglib-ocaml libvorbis-ocaml lsb-release \
monit mpg123 multitail odbc-postgresql patch php-pear php5-curl php5-gd \
php5-pgsql postgresql python2.6 python-virtualenv rabbitmq-server sudo \
tar vorbis-tools
```

The other dependency to install is *phing* (note that you must use the version number shown):

```
$ sudo pear channel-discover pear.phing.info
$ sudo pear install phing/phing-2.4.2
```

2. Check that the Apache web server modules that Airtime requires are enabled:

```
$ sudo a2enmod php5 rewrite
```

The server should respond:

```
Module php5 already enabled
Module rewrite already enabled
```

3. Create a directory to contain the Airtime web interface:

```
$ sudo mkdir -p /usr/share/airtime/public
```

4. Next, create the Airtime virtual host configuration file for Apache:

```
$ sudo nano /etc/apache2/sites-available/airtime
```

and enter the information below, substituting your server's hostname for *airtime.example.com* and your system administrator's email address for *admin@example.com*. Make sure you set the *DocumentRoot* and *Directory* paths correctly. This should match the *public* directory that the installer will unpack the web interface into, which by default is the */usr/share/airtime/public/* directory.

```
<VirtualHost *:80>
   ServerName airtime.example.com
   ServerAdmin admin@example.com
   DocumentRoot /usr/share/airtime/public
   php_admin_value upload_tmp_dir /tmp

  <Directory /usr/share/airtime/public>
      DirectoryIndex index.php
      AllowOverride all
      Order allow,deny
      Allow from all
  </Directory>
</VirtualHost>
```

Press Ctrl+O to save the file, then Ctrl+X to exit the **nano** editor.

5. Create the PHP configuration file */etc/airtime/airtime.ini* in nano:

```
$ sudo nano /etc/airtime/airtime.ini
```

with the following contents:

```
[PHP]
memory_limit = 512M
magic_quotes_gpc = Off
file_uploads = On
upload_tmp_dir = /tmp
```

Save and exit nano, then link this file to the system's PHP configuration with the command:

```
$ sudo ln -s /etc/airtime/airtime.ini /etc/php5/conf.d/airtime.ini
```

6. Enable the new configuration by entering the command:

```
$ sudo a2ensite airtime
```

The server should respond:

```
Enabling site airtime.
Run '/etc/init.d/apache2 reload' to activate new configuration!
```

You may also need to disable the default site configuration, which may otherwise interfere with your Airtime installation:

```
$ sudo a2dissite default
```

As suggested by the output of the command above, reload the web server configuration.

```
$ sudo /etc/init.d/apache2 reload
```

The server should respond:

```
 * Reloading web server config apache2
```

7. Download Airtime from https://sourceforge.net/projects/airtime/files/ with **wget**. For example, to download version 2.0.0, you could use the command:

```
$ wget http://downloads.sourceforge.net/project/airtime/2.0.0/airtime-2.0.0.tar.gz
```

8. Unzip the downloaded file in your home directory. This action will create a subdirectory called *airtime-2.0.0*:

```
$ sudo tar -xvzf airtime-2.0.0.tar.gz -C ~/
```

9. Monit is a utility which Airtime uses to make sure that the system runs smoothly. Enable it by opening the */etc/default/monit* file in the **nano** editor:

```
$ sudo nano /etc/default/monit
```

Find the line that begins with *startup* and change the value to 1:

```
startup=1
```

Save the file with Ctrl+O and close nano with Ctrl+X. Now copy the Monit configuration from the Airtime install directory to the */etc/monit/conf.d/* directory:

```
$ sudo cp ~/airtime-2.0.0/python_apps/monit/airtime-monit.cfg /etc/monit/conf.d/
```

Open the */etc/monit/monitrc* file in **nano**:

```
$ sudo nano /etc/monit/monitrc
```

At the end of the file, add the line:

```
include /etc/monit/conf.d/*
```

Save the file with Ctrl+O and close nano with Ctrl+X. Then start Monit with:

```
$ sudo invoke-rc.d monit start
```

More information about monit is available in the chapter *Using Monit*.

10. On Debian squeeze, make sure the rabbitmq-server daemon has started:

```
$ sudo invoke-rc.d rabbitmq-server start
```

11. Finally, run the **airtime-install** script:

```
$ sudo ~/airtime-2.0.0/install_minimal/airtime-install
```

Once the **airtime-check-system** script confirms that the install has been successful, you are now ready to proceed to the *Configuration* chapter.

INSTALL SCRIPT OPTIONS

By default, the **airtime-install** script preserves any existing configuration or installation that it finds on the server. However, it is also possible to dictate the behaviour of the script with a command line option, as follows:

```
--help|-h            Displays usage information.
--overwrite|-o       Overwrite any existing config files.
--preserve|-p        Keep any existing config files.
--no-db|-n           Turn off database install.
--reinstall|-r       Force a fresh install of this Airtime version
```

MANUAL UNINSTALL

To manually uninstall Airtime from the server, run the **airtime-uninstall** script from the original installation directory, for example:

```
$ sudo ~/airtime-2.0.0/install_minimal/airtime-uninstall
```

Optionally, you can also delete the Airtime storage archive and configuration folders, if you have backups and are not going to need the data on this particular server again. The **rm** command should be used with caution, because it has no undo feature.

```
$ sudo rm -r /srv/airtime
$ sudo rm -r /etc/airtime
```

20. AUTOMATED INSTALLATION

This installation method is intended for computers running Ubuntu or Debian GNU/Linux, and is the recommended method for production Airtime systems. If you have previously performed a manual installation of Airtime on the server, you should run the *airtime-uninstall* script to remove it before setting up the server for automated installation.

SET UP REPOSITORIES FOR UBUNTU

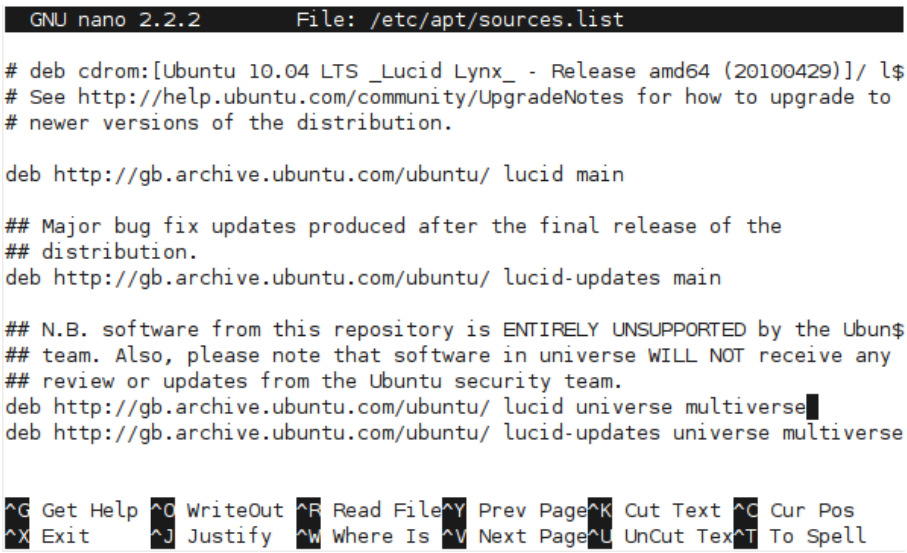
When installing on an Ubuntu server, a few of the packages that Airtime relies on are in the Ubuntu *universe* or *multiverse* repositories. If either of these repositories is disabled, you can enable them in the */etc/apt/sources.list* file, by opening the *nano* editor in your server's console. The *nano* editor should be installed by default, but if not, you can install it with the command:

```
$ sudo apt-get install nano
```

Then open the *sources.list* file with the command:

```
$ sudo nano /etc/apt/sources.list
```

For an Ubuntu Lucid server, find the lines which begin *deb* and end with *lucid universe* or *lucid-updates universe*, adding *multiverse* to the end of these lines, if it is not there already. The *multiverse* repository is required for the *libmp3lame0* library, which is an MP3 encoder.

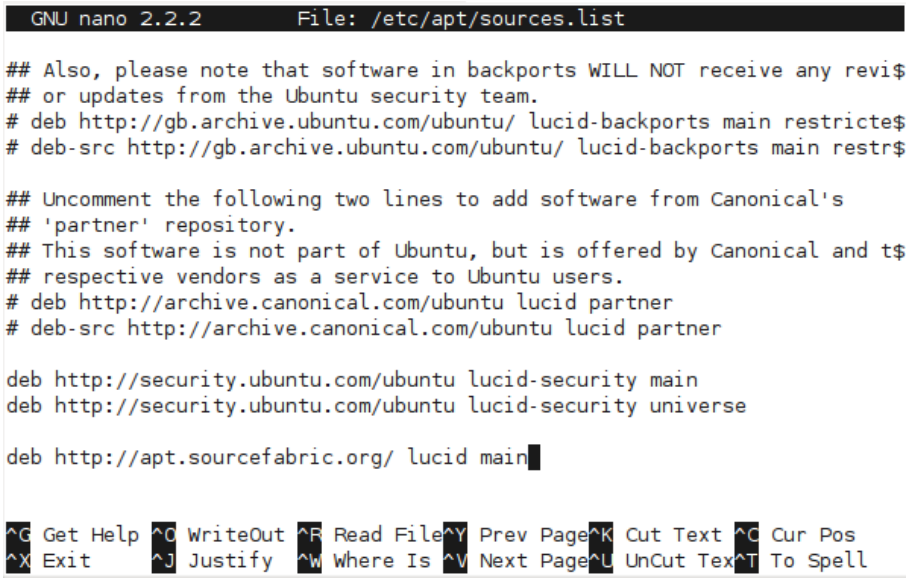


The exact repository mirror URLs in your *sources.list* file will differ from the above screenshot, depending on your location.

The Sourcefabric repository contains packages for Airtime, and any other packages which Airtime requires. To add the Sourcefabric repository to an Ubuntu Lucid server, scroll to the end of the *sources.list* file and add the following line:

```
deb http://apt.sourcefabric.org/ lucid main
```

For Ubuntu Maverick, Natty or Oneiric, substitute *maverick, natty* or *oneiric* in place of *lucid* in the line above.



Press Ctrl+O (the Ctrl key and the letter O together) to save the file, then Ctrl+X to exit the nano editor.

SET UP REPOSITORIES FOR DEBIAN

On a Debian squeeze server, you can edit the */etc/apt/sources.list* file as root with the command:

```
# nano /etc/apt/sources.list
```

Packages for MP3 encoding are not included in the Debian squeeze repositories. You can obtain the necessary *libmp3lame0* package by adding the following repository to the end of the file:

```
deb http://backports.debian.org/debian-backports squeeze-backports main
```

To add the Sourcefabric repository to a Debian squeeze server, add the following line to the end of the file:

```
deb http://apt.sourcefabric.org/ squeeze main
```

Press Ctrl+O (the Ctrl key and the letter O together) to save the file, then Ctrl+X to exit the *nano* editor.

INSTALL THE SOURCEFABRIC SIGNING KEY

Reload the system's package list with:

```
$ sudo apt-get update
```

You will see an error message about a missing public key.

```
Hit http://gb.archive.ubuntu.com lucid Release
Hit http://gb.archive.ubuntu.com lucid-updates Release
Hit http://gb.archive.ubuntu.com lucid/main Packages
Hit http://gb.archive.ubuntu.com lucid/universe Packages
Hit http://gb.archive.ubuntu.com lucid/multiverse Packages
Hit http://gb.archive.ubuntu.com lucid-updates/main Packages
Hit http://gb.archive.ubuntu.com lucid-updates/universe Packages
Hit http://gb.archive.ubuntu.com lucid-updates/multiverse Packages
Fetched 491B in 2s (218B/s)
Reading package lists... Done
W: GPG error: http://apt.sourcefabric.org lucid Release: The following si
gnatures couldn't be verified because the public key is not available: NO
_PUBKEY 0888FE5B174C1854
daniel@office:~$
```

To fix this system error, you need to install the *sourcefabric-keyring* package, which contains the package signing key. This encryption key is a security measure which helps ensure that the Airtime packages you will be downloading in future have not been tampered with by a third party. You can install the key with the command:

```
$ sudo apt-get install sourcefabric-keyring
```

When prompted, press the *y* key on your keyboard to install the *sourcefabric-keyring* package without verification. If you wish to verify the authenticity of the package signing key, please contact Sourcefabric for assistance.

```
daniel@office:~$ sudo apt-get install sourcefabric-keyring
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed
  sourcefabric-keyring
0 upgraded, 1 newly installed, 0 to remove and 7 not upgraded.
Need to get 3,610B of archives.
After this operation, 49.2kB of additional disk space will be used.
WARNING: The following packages cannot be authenticated!
  sourcefabric-keyring
Install these packages without verification [y/N]? y
Get: 1 http://apt.sourcefabric.org/ lucid/main sourcefabric-keyring 0.1 [3
,610B]
Fetched 3,610B in 0s (12.3kB/s)
Selecting previously deselected package sourcefabric-keyring.
(Reading database ... 348822 files and directories currently installed.)
Unpacking sourcefabric-keyring (from .../sourcefabric-keyring_0.1_all.deb)
 ...
Setting up sourcefabric-keyring (0.1) ...

daniel@office:~$
```

Update your computer's software sources again, to confirm that you are now using a trusted software repository:

```
$ sudo apt-get update
```

You should no longer see the error message about the missing public key.

INSTALL THE DATABASE MANAGEMENT SYSTEM

Airtime uses a PostgreSQL database to keep track of media assets and associated metadata in its storage server. Depending on the scale of your Airtime installation and the hardware available, you may wish to install PostgreSQL on a separate server. If you only have one server, you can install the *postgresql* package on the same machine as Airtime with the command:

```
$ sudo apt-get install postgresql
```

INSTALL A STREAMING SERVER

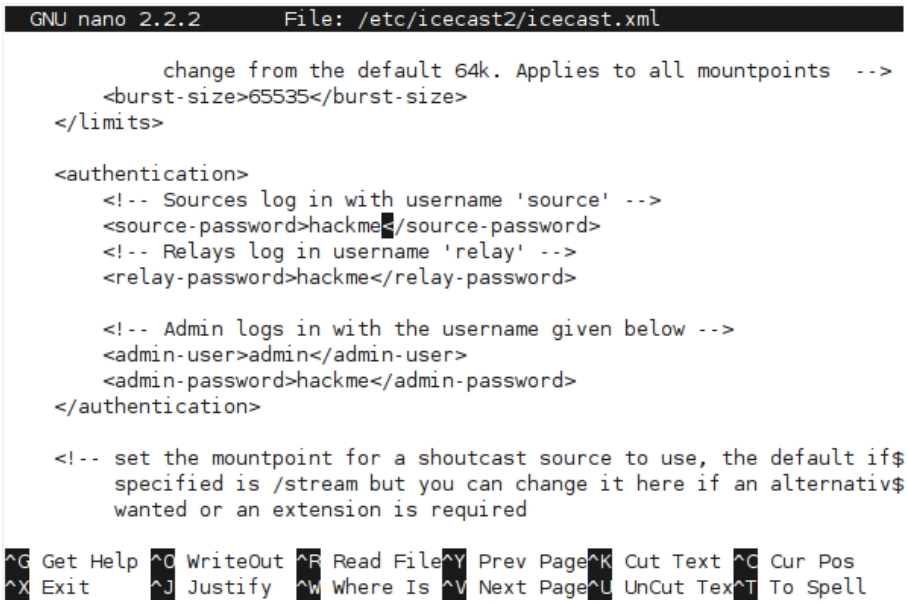
Optionally, you may wish to stream directly from Airtime to an **Icecast** media distribution server, without requiring a soundcard or mixer in the broadcast chain. This option is particularly suitable for fully automated stations, in which all shows are played out using Airtime. You can install the *icecast2* package on your server with the command:

```
$ sudo apt-get install icecast2
```

In some scenarios, you may wish to stream from Airtime to a remote Icecast server, for example when Icecast is installed on a server in a datacenter with greater bandwidth available than an Airtime server located at your broadcast studio has. This separation may become necessary if your stream becomes popular, because each additional listener which connects to the Icecast stream uses additional bandwidth. In this case, you do not need to install the *icecast2* package on the Airtime server.

Before running Icecast for the first time, you should edit the file */etc/icecast2/icecast.xml* to change the default *<source-password>*, *<relay-password>* and *<admin-password>* values from '*hackme*' to something more secure.

```
$ sudo nano /etc/icecast2/icecast.xml
```



You should also set the value of *<hostname>* to the domain name of the Icecast server, for example:

```
<hostname>icecast.example.com</hostname>
```

Save and close the *icecast.xml* file with Ctrl+O and Ctrl+X. Then set the Icecast server to start automatically when the server boots in the */etc/default/icecast2* file:

```
$ sudo nano /etc/default/icecast2
```

by setting the value of *ENABLE* to *true* on the last line of that file:

```
ENABLE=true
```

Save and close this file with Ctrl+O and Ctrl+X, then start Icecast:

```
$ sudo service icecast2 start
```

The server should respond:

```
Starting icecast2: Starting icecast2
Detaching from the console
icecast2.
```

INSTALL AIRTIME

You can now install the Airtime package with:

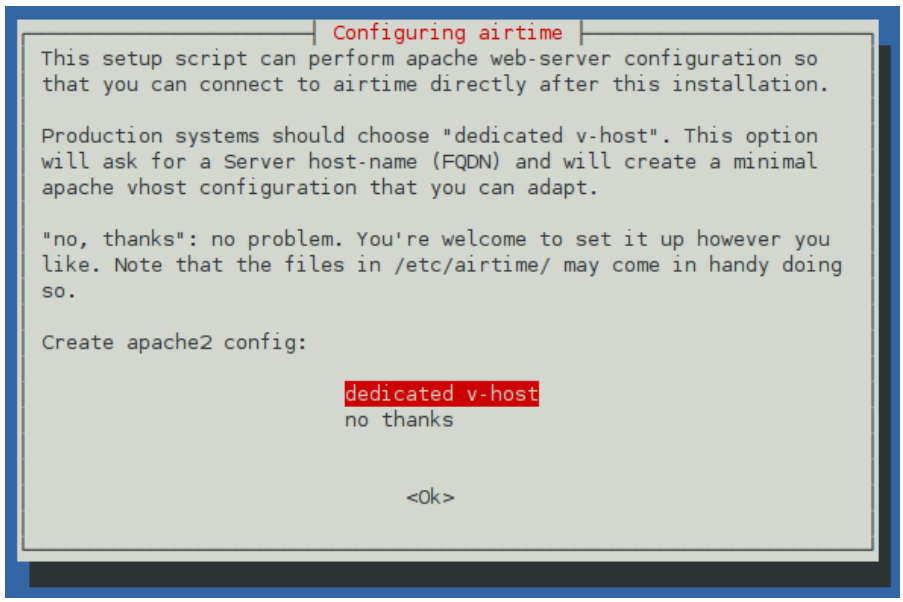
```
$ sudo apt-get install airtime
```

This command will install all of the Airtime components, plus any other packages that Airtime requires in order to run.

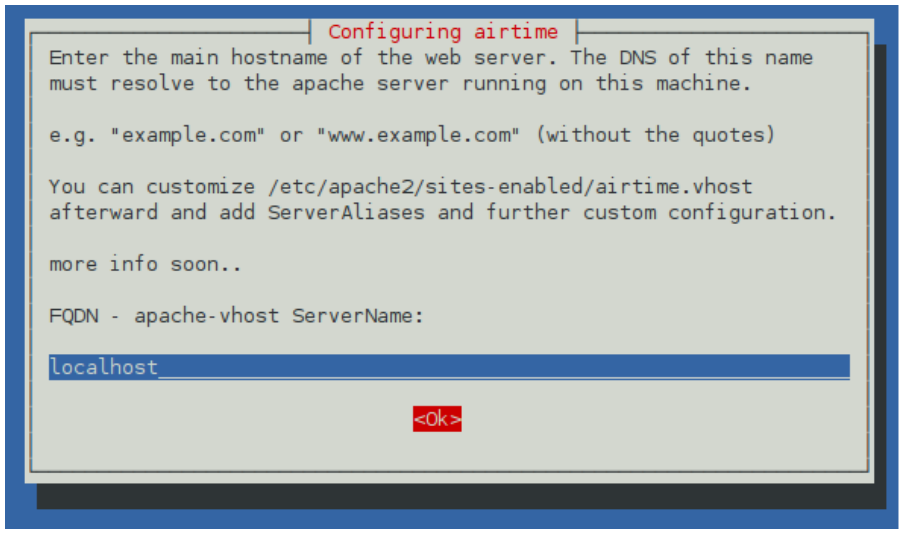
```
$ sudo apt-get install airtime
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  airtime-audio-samples alsa-utils
The following NEW packages will be installed
  airtime liquidsoap
0 upgraded, 2 newly installed, 0 to remove and 0 not upgraded.
Need to get 0B/11.4MB of archives.
After this operation, 62.3MB of additional disk space will be used.
Preconfiguring packages ...
Selecting previously deselected package airtime.
(Reading database ... 400129 files and directories currently installed.)
Unpacking airtime (from .../airtime_2.0.0-3_all.deb) ...
```

Once all the packages have been downloaded and installed, you will be asked a few questions about the configuration of the Airtime system. You can accept the default settings by pressing the **Enter** key, or use the **Tab** key to change your selection.

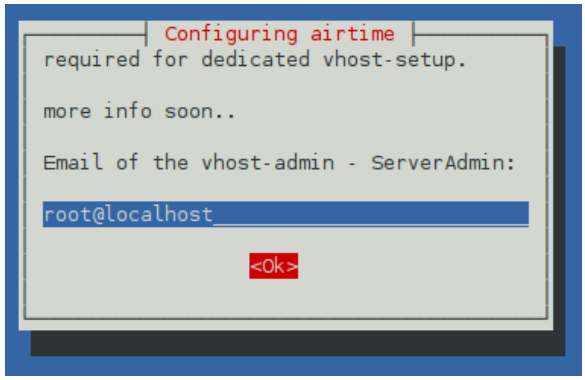
Firstly, you will asked if you wish to create an **Apache** virtual host web server configuration automatically.



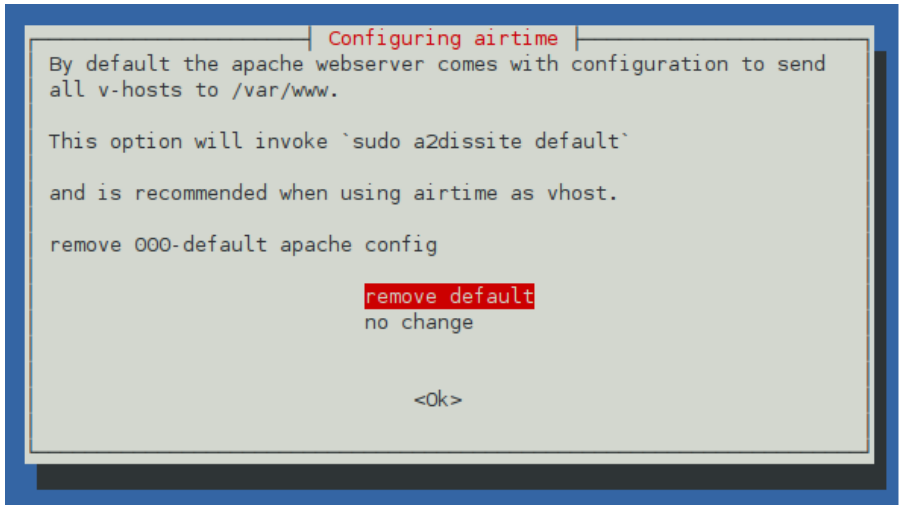
If so, you will need to enter the domain name that your station staff will use to access Airtime. For a test server that will only be accessed directly from the same machine, you can use the default setting of *localhost* here.



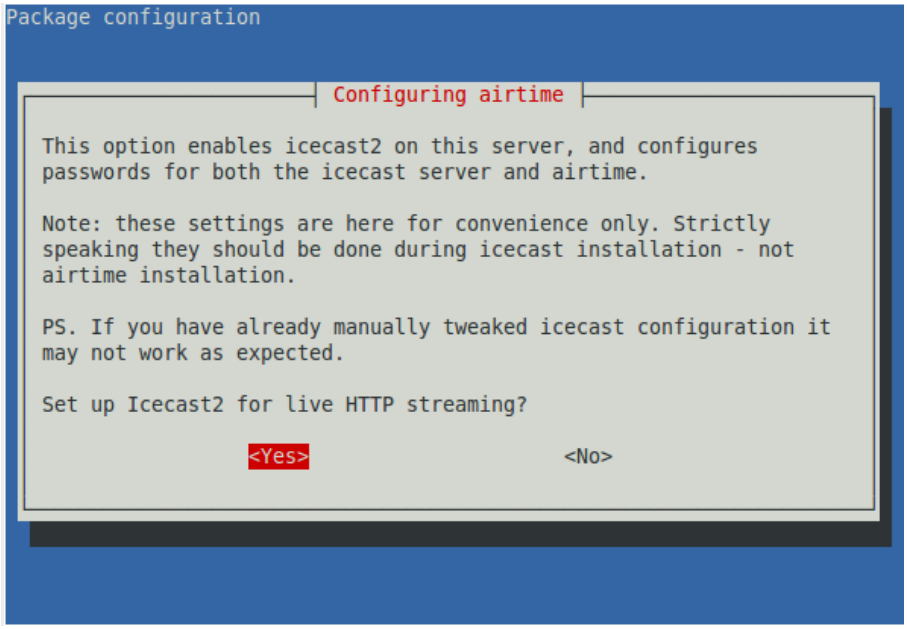
Next, set the contact email address of the server administrator, which is good practice in case of any server problems. For a test server, using an address at the *localhost* is acceptable.



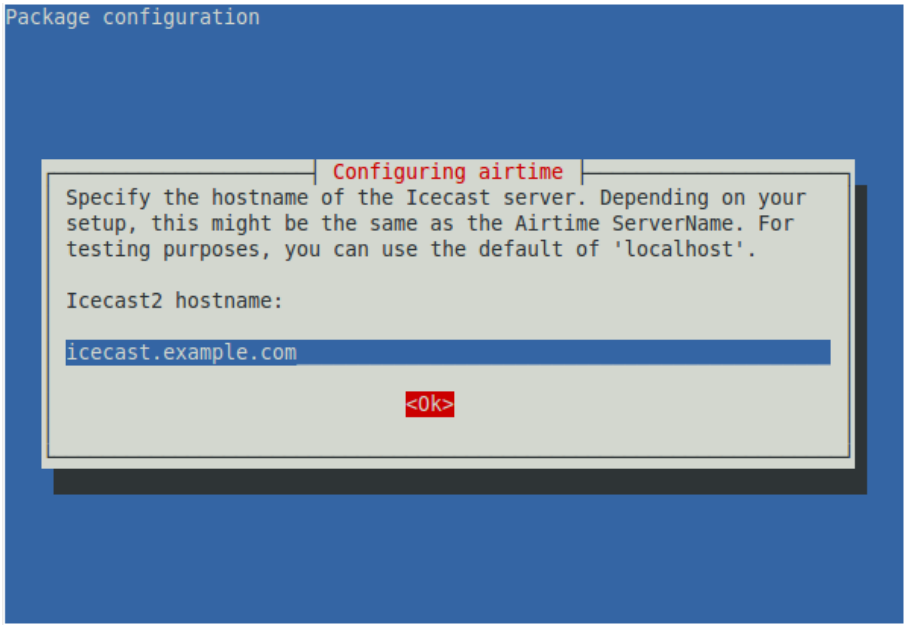
Debian and Ubuntu servers are set up with a default Apache configuration, which might block station staff from accessing Airtime. If you wish, this default configuration can be removed automatically for you.



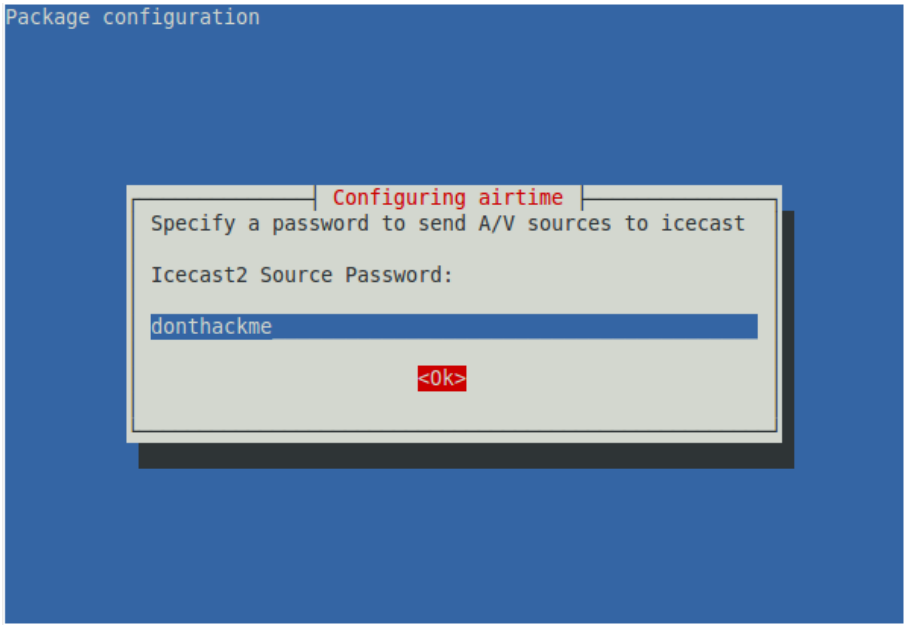
If you are setting up Airtime to stream directly to Icecast without using a soundcard in between, you can set the hostname and passwords for the Icecast server during the Airtime installation.



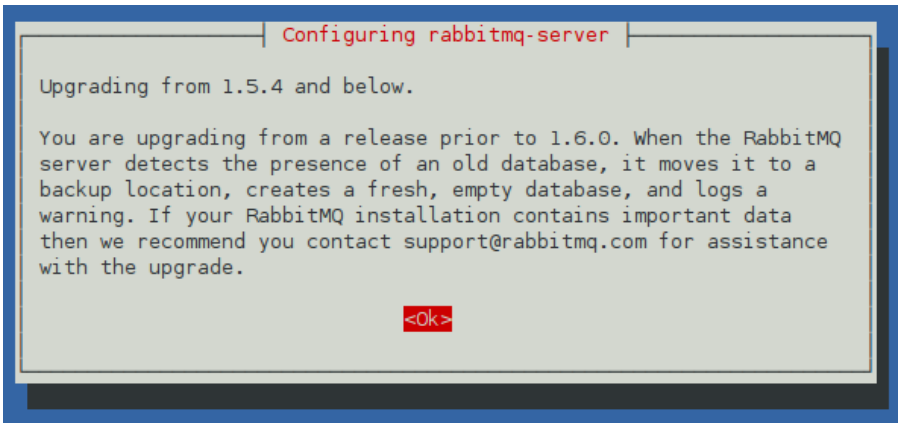
First, set the hostname of the Icecast server. If you have Icecast installed on the same machine as Airtime, you can use *localhost* here.



The security of your streaming server depends on the strength of the passwords that you choose. You should set strong passwords for *source*, *relay* and *admin* access.



On some GNU/Linux distributions, you may be warned about upgrading the *rabbitmq-server* package, even if you have never installed this package before. If RabbitMQ is only being used by Airtime on your server, it is safe to press the **Enter** key to continue.



The Airtime installer will then run a script to check that your server environment is set up correctly.

```
*** Verifying your system environment, running airtime-check-system ***

AIRTIME_STATUS_URL              = http://localhost/api/status/format/json/api_key/%%api_key%%
AIRTIME_SERVER_RESPONDING       = OK
KERNEL_VERSION                  = 2.6.32-37-generic
MACHINE_ARCHITECTURE            = x86_64
TOTAL_MEMORY_MBYTES             = 6128244
TOTAL_SWAP_MBYTES               = UNKNOWN
AIRTIME_VERSION                 = 2.0.0
OS                              = Ubuntu 10.04.3 LTS x86_64
CPU                             = Dual Core AMD Opteron(tm) Processor 170
WEB_SERVER                      = Apache/2.2.14 (Ubuntu)
PLAYOUT_ENGINE_PROCESS_ID       = 14317
PLAYOUT_ENGINE_RUNNING_SECONDS  = 72
PLAYOUT_ENGINE_MEM_PERC         = 0.2%
PLAYOUT_ENGINE_CPU_PERC         = 0.0%
LIQUIDSOAP_PROCESS_ID           = 14314
LIQUIDSOAP_RUNNING_SECONDS      = 72
LIQUIDSOAP_MEM_PERC             = 0.4%
LIQUIDSOAP_CPU_PERC             = 17.4%
MEDIA_MONITOR_PROCESS_ID        = 14242
```

```
MEDIA_MONITOR_RUNNING_SECONDS  = 73
MEDIA_MONITOR_MEM_PERC         = 0.2%
MEDIA_MONITOR_CPU_PERC         = 0.4%
SHOW_RECORDER_PROCESS_ID       = 14343
SHOW_RECORDER_RUNNING_SECONDS  = 72
SHOW_RECORDER_MEM_PERC         = 0.2%
SHOW_RECORDER_CPU_PERC         = 0.0%
RABBITMQ_PROCESS_ID            = 1623
RABBITMQ_RUNNING_SECONDS       = 70
RABBITMQ_MEM_PERC              = 0.3%
RABBITMQ_CPU_PERC              = 0.0%

-- Your installation of Airtime looks OK!

*************** Install Complete ***************
```

You are now ready to proceed to the *Configuration* chapter.

21. CONFIGURATION

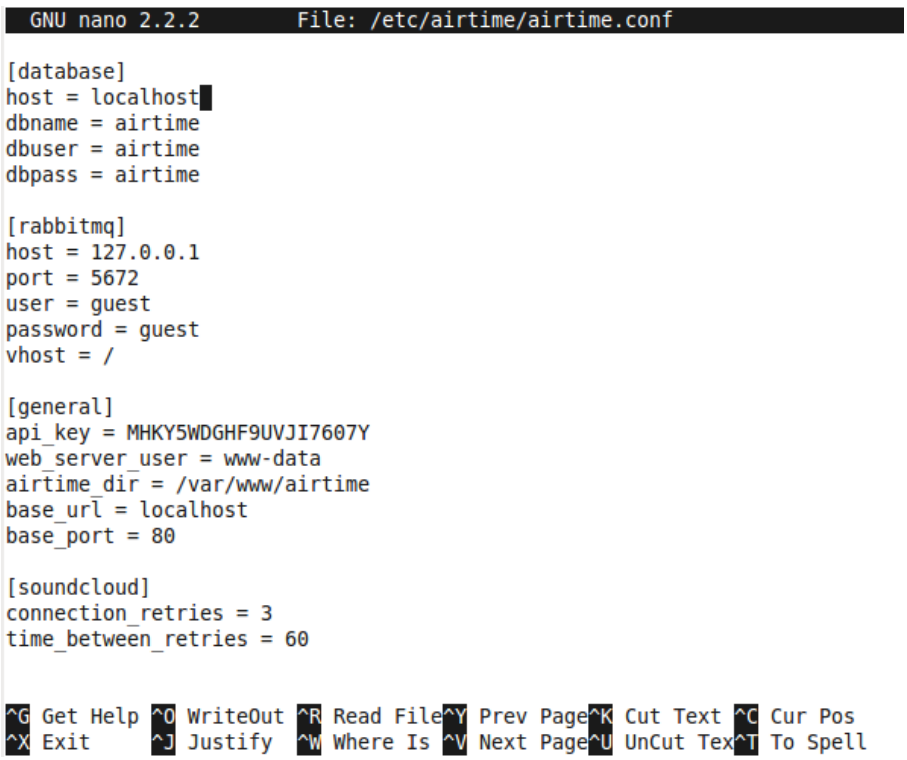
The broadcast playout configuration for Airtime is shown in the file */etc/airtime/liquidsoap.cfg* which is automatically generated by the **Stream Settings** page of the Airtime administration interface. For this reason, you should not edit the configuration manually, as any changes are likely to be overwritten by the administration interface.

ADVANCED SETTINGS

Optionally, you may wish to edit the file */etc/airtime/airtime.conf* to set the PostgreSQL database host, and the username and password to connect to the database with:

```
$ sudo nano /etc/airtime/airtime.conf
```

You can also set options for RabbitMQ messaging, the Airtime server and SoundCloud uploads in this file, although you should not normally need to adjust the defaults.



Save and close the file with Ctrl+O and Ctrl+X. If you have changed the database settings, you should now run the command:

```
$ sudo airtime-update-db-settings
```

to make sure all of Airtime's database configuration files are updated. This command should output a large amount of text to the server console, ending with:

```
BUILD FINISHED
```

and a report of the total time taken to update the configuration.

You should now be able to log in to the Airtime administration interface, as shown in the **Getting started** chapter.

22. SETTING THE SERVER TIME

Accurate time keeping on your server is vital for optimal Airtime performance. You can confirm that the date and time of your server are set correctly with the **date** command:

```
$ date
```

The server should respond with the date, time, time zone and year in a format similar to the following example:

```
Wed Jan  4 10:54:47 GMT 2012
```

The time zone reported should match the continent and city set in the **Preferences** page of the Airtime administration interface, such as *GMT* in the example above matching *Europe/London*. If not, see the section **Adjusting the server time zone** below.

CONFIGURING NTP

Although it is possible to set the date and time of the server manually, this is not recommended because the server clock can drift over time, compromising the accuracy of your broadcast schedule. If your Airtime server is permanently connected to the Internet, you can synchronize your server to a time server with the **ntp** program. If **ntp** is not yet installed, you can enter the following command on Debian or Ubuntu:

```
$ sudo apt-get install ntp
```

Optionally, open the **ntp** configuration file in the **nano** editor to add further time server names:

```
$ sudo nano /etc/ntp.conf
```

On Ubuntu GNU/Linux, the default time server is *ntp.ubuntu.com*, but there are many other time servers available on the public Internet, including the group of servers listed at http://www.pool.ntp.org/ for each country. Using a variety of NTP servers located closely to your Airtime server should produce the most accurate results. For example, for a server in the United Kingdom you could use the following list:

```
# You do need to talk to an NTP server or two (or three).
server ntp.ubuntu.com
server 0.uk.pool.ntp.org
server 1.uk.pool.ntp.org
server 2.uk.pool.ntp.org
server 3.uk.pool.ntp.org
```

Enter the server names you require, press Ctrl+O to write out the */etc/ntp.conf* file, then Ctrl+X to exit **nano**. Restart the **ntp** service with:

```
$ sudo invoke-rc.d ntp restart
```

The server should respond:

```
* Stopping NTP server ntpd                                  [ OK ]
* Starting NTP server ntpd                                  [ OK ]
```

Then use the **ntpq -p** command to confirm that **ntp** is working. This command should produce output similar to the following:

```
daniel@airtimebox:~$ ntpq -p
     remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
 europium.canoni 193.79.237.14   2 u   28   64    3   39.571   12.600   3.590
 norb.v4.colo.m. 46.227.200.72   3 u   28   64    3   47.856   -6.908  10.028
 82.113.154.206  193.62.22.82    2 u   29   64    3   11.458   -0.513   2.629
```

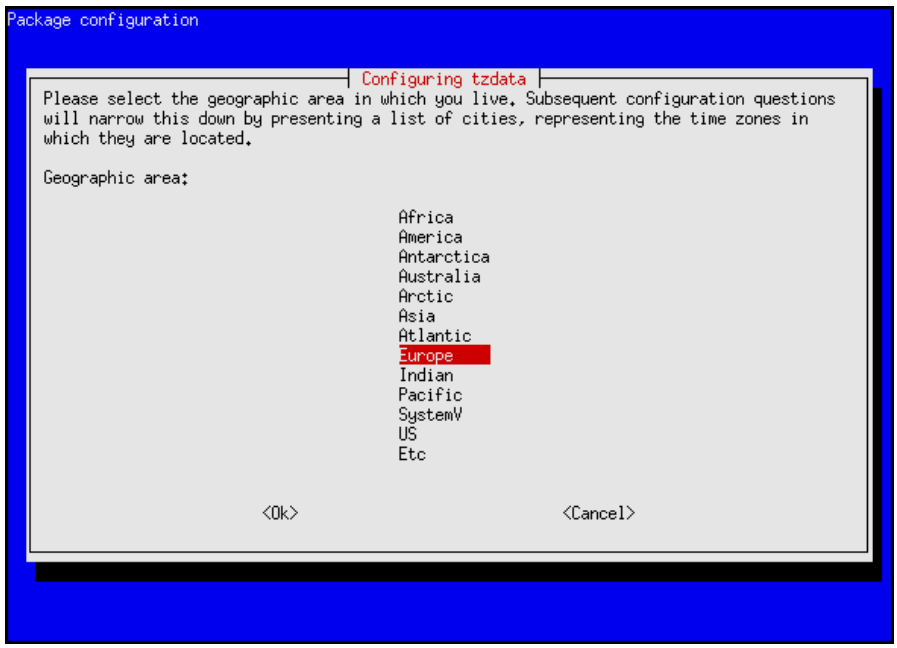
```
ntppub.le.ac.uk 158.43.192.66   2 u   91   64    2 122.781  44.864   0.001
dns0.rmplc.co.u 195.66.241.3    2 u   27   64    3  22.171   1.464   4.242
```

ADJUSTING THE SERVER TIME ZONE

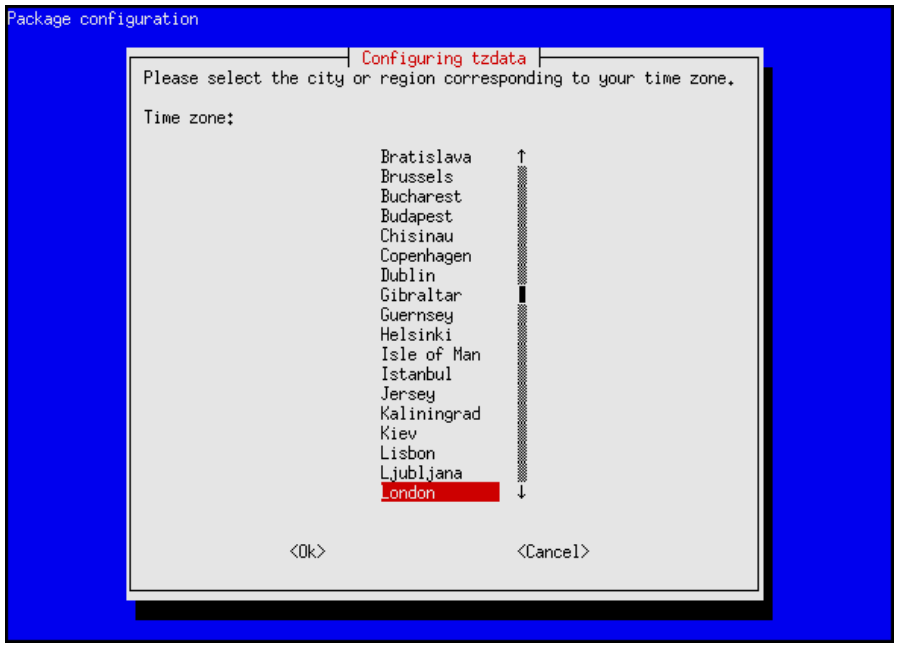
If your server is set to a different time zone than the one that you set during Airtime installation, your broadcast schedule may be played out at incorrect times. The data center which hosts your Airtime server could be located anywhere in the world. Some servers are set to Coordinated Universal Time or UTC (similar to Greenwich Mean Time or GMT), regardless of their location. If this is not appropriate for your station, on a Debian or Ubuntu server you can reconfigure the **tzdata** (time zone data) package with the command:

```
$ sudo dpkg-reconfigure tzdata
```

This command opens a menu in which you can select the continent that you require, by pressing the Enter key.



The next step is to select your nearest city, again by pressing the Enter key. The appropriate time zone is selected according to the information that you have entered.



The console output from the **dpkg-reconfigure tzdata** command will confirm the new setting:

```
Current default time zone: 'Europe/London'
Local time is now:      Mon Mar  7 15:04:58 GMT 2011.
Universal Time is now:  Mon Mar  7 15:04:58 UTC 2011.
```

ADMINISTRATION

23. USING THE IMPORT SCRIPT

If you have a large number of files in your media library, importing these files one at a time into a broadcast automation system would be time-consuming and tedious. That's why Airtime includes a script that can import an entire directory of files in one go.

METADATA QUALITY

The **airtime-import** script automatically imports any metadata that is in the files' ID3 tags. If these tags are incorrect or are missing information, you will have to either edit the metadata before importing them, or suffer the consequences. For example, if the tags have creator or genre metadata missing, it will be impossible to search for, playlist or schedule the media according to these criteria.

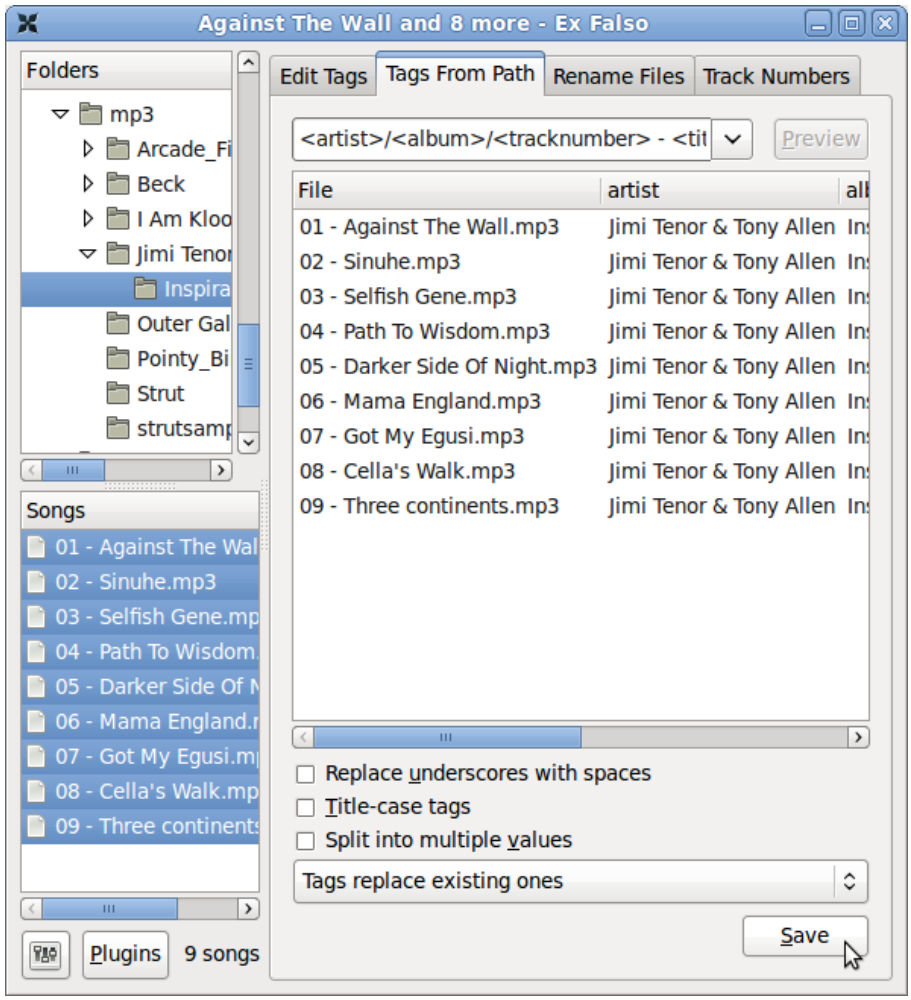
There are a number of programs available which can be used to correct mistakes or incomplete information in ID3 tags. On GNU/Linux, the program **Ex Falso** (http://code.google.com/p/quodlibet/) can be useful for batch setting and editing ID3 tags before importing files into your Airtime server. On an Ubuntu desktop machine, you can install this program with the command:

```
$ sudo apt-get install exfalso
```

After installation, you can run the program with the command:

```
$ exfalso
```

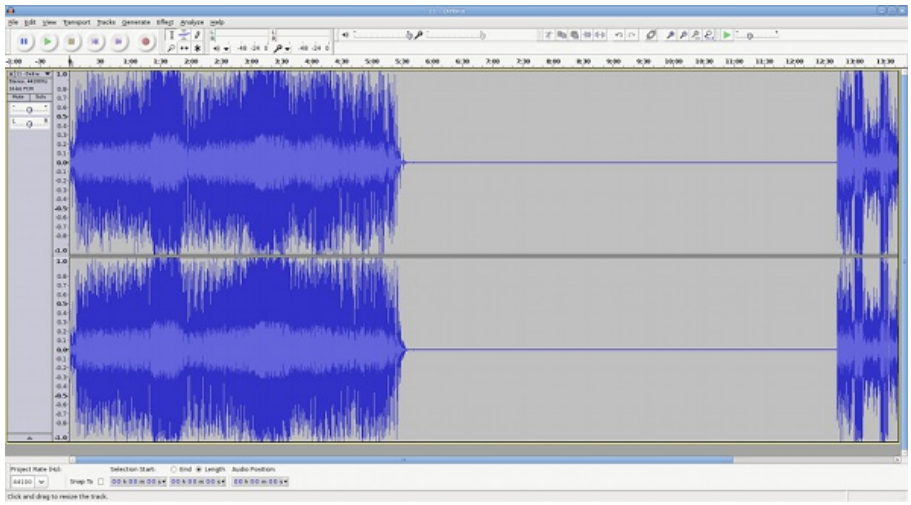
The *Tags From Path* feature of this program is a particularly useful time saver if you have a large archive of untagged files. Sometimes there is useful creator or title information in the file name or directory path structure, which can be converted into an ID3 tag automatically.



SILENCE IN MEDIA FILES

Before importing media, it is good practice to check for any silent sections in the media files. While Airtime can compensate for leading and trailing silence with the use of cue-in and cue-out points in the **Playlist Builder**, it is better to trim these files to the intended length before upload. This is because trimmed files do not require station staff to set cue points over and over again, as media in the Airtime storage archive could potentially be re-used for many different shows. **Audacity** is a cross-platform editor suitable for the task of trimming audio files, available from http://audacity.sourceforge.net/

Very quiet introductions or over-long fades can also lead to apparent gaps in your broadcast playout. Some audio CDs feature a 'hidden track' at the end, which in fact uses a long period of silence within the final track, rather than an actual separate track on the disc. This means that CD encoding programs will encode both the hidden material and the silence in the media file. For example, the track *Debra* from the CD *Midnite Vultures* by *Beck* includes hidden material preceded by seven minutes of silence, as shown in the screen shot from Audacity below.



COPYING VERSUS WATCHING

There are three main options when using the **airtime-import** script: Either to **copy** or **move** files into Airtime's main storage folder, or to **watch** files elsewhere. Each option has advantages and disadvantages, so you will have to think about how the files will be used in your station in the long term.

- For files that are *copied* or *moved*, you may run into problems with hard disk space if you do not remove files from the storage server periodically.
- For files that are *watched*, you must make sure that the files are actually going to be there at playout time, otherwise they will be skipped. For this reason, it is not recommended to use the **watch** option for easily-removable storage (like MP3 players or USB memory sticks). There's a strong likelihood that the storage might be unplugged and taken away before playout time. It is also important to make sure that any external hard disk with watched files is powered on, and stays on.

The **airtime-import** script works from the command line, so you must first log into a console on the Airtime server. In recent versions of Airtime, you no longer need to specify the full file system path to the media files that you wish to import. Copying is specified with the **copy** sub-command:

```
$ airtime-import copy <directory>
```

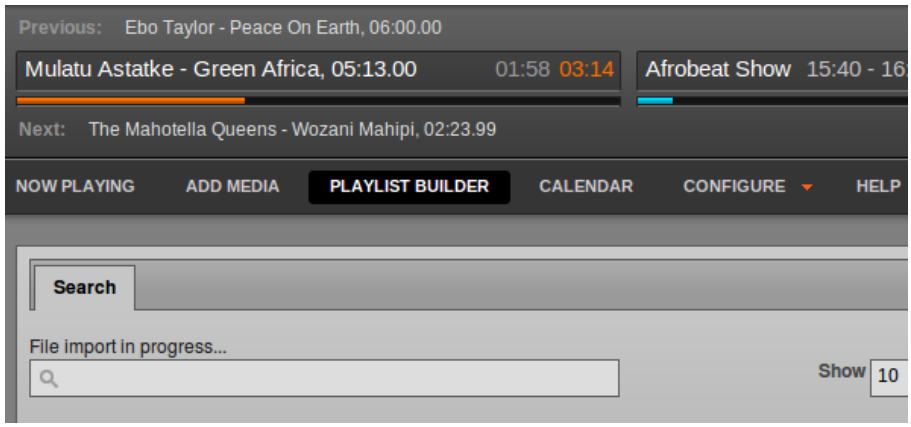
where *<directory>* is the root directory of the media files that you wish to import.

Alternatively, you may wish to delete the files from the original location after copying them to the storage archive, which is accomplished with the **move** sub-command:

```
$ airtime-import move <directory>
```

If the new files are exact duplicates of files that have already been imported, the script will detect this.

Any users logged into the administration interface will be warned if a file import is in progress. Users assigned the *admin* privilege should not attempt to change Airtime's default storage directory while an import is taking place.



Watching a new folder is specified by using the **watch add** sub-command:

```
$ airtime-import watch add <directory>
```

The script will report the name of the folder now being watched, for example:

```
$ airtime-import watch add /home/daniel/Music/mp3/Beck
/home/daniel/Music/mp3/Beck/ added to watched folder list successfully
```

The watched folder will also be listed on the **Manage Media Folders** page in the Airtime administration interface. To obtain a list of currently watched folders in the server console, you can use the **watch list** sub-command:

```
$ airtime-import watch list
/home/daniel/Music/mp3/Beck/
```

Any new files copied to a watched folder (for example, using your desktop computer's file manager) will automatically appear in the Airtime database, and are then ready for playout. In the same way, if you delete a media file using your file manager, it will be automatically removed from the Airtime database, and will no longer be available for playout.

If you wish to no longer watch a particular folder, use the **watch remove** sub-command:

```
$ airtime-import watch remove /home/daniel/Music/mp3/Beck
/home/daniel/Music/mp3/Beck/ removed from watched folder list successfully
```

GET OR SET THE STORAGE FOLDER PATH

The airtime-import script also enables you to read or write Airtime's storage folder path configuration. You can find out the current setting with the **storage-dir get** sub-command:

```
$ airtime-import storage-dir get
/srv/airtime/stor
```

A new storage folder path can be configured using the **storage-dir set** sub-command:

```
$ airtime-import storage-dir set /home/daniel/Music/mp3/
Successfully set storage folder to /home/daniel/Music/mp3/
```

The **storage-dir set** sub-command should be used with caution on a production server, because moving the storage folder during scheduled programming has the potential to disrupt your broadcast playout.

24. THE AIRTIME-USER COMMAND

Airtime includes a command-line tool **airtime-user** which can be used to add, update or remove user accounts. You can see the options for the command by typing it in the server console without any arguments:

```
$ airtime-user

airtime-user
===============
    This program allows you to manage Airtime users.

OPTIONS:
    --addupdate <username>
        Add the user or update user information.
    --delete <username>
        Remove the user.
```

For example, to create an account for a DJ known as *Disco Dave*, you could enter the following:

```
$ airtime-user --addupdate dave
Creating user
Enter password (min 6 characters): saturdaynightfever1977
Enter first name: Disco
Enter last name: Dave
Enter user type [(A)dmin|(P)rogram Manager|(D)J|(G)uest]: D
```

Existing user accounts can be updated using the same command.

25. THE AIRTIME-LOG COMMAND

The airtime-log command provides convenient access to the logging output from the services which make up the Airtime system: *media-monitor*, *recorder*, *playout*, *liquidsoap* and *web*.

Using this command requires root privileges (**sudo** on Ubuntu). Entering the command without any options returns a list of options that you can specify:

```
$ sudo airtime-log

Usage: airtime-log [options]

--view|-v <string>     Display log file
        media-monitor|recorder|playout|liquidsoap|web

--dump|-d [ <string> ] Collect all log files and compress into a tarball
        media-monitor|recorder|playout|liquidsoap|web (ALL by default)

--tail|-t [ <string> ] View any new entries appended to log files in real-time
        media-monitor|recorder|playout|liquidsoap|web (ALL by default)
```

For example, to view the media-monitor log, you could use the command:

```
$ sudo airtime-log -v media-monitor
```

The server console will display something like the following output:

```
2011-11-29 09:32:45,750 INFO - [MainThread] [MediaMonitor.py : <module>()] : LINE 32 -

*** Media Monitor bootup ***

2011-11-29 09:32:45,864 INFO - [MainThread] [MediaMonitor.py : <module>()] : LINE 39 - Setting
up monitor
2011-11-29 09:32:46,058 INFO - [MainThread] [api_client.py : setup_media_monitor()] : LINE 405
- Connected to Airtime Server. Json Media Storage Dir: {u'watched_dirs': [], u'stor':
u'/srv/airtime/stor/'}
2011-11-29 09:32:51,061 INFO - [MainThread] [MediaMonitor.py : <module>()] : LINE 47 - Storage
Directory is: /srv/airtime/stor/
2011-11-29 09:32:51,066 INFO - [MainThread] [MediaMonitor.py : <module>()] : LINE 54 -
Initializing event processor
2011-11-29 09:32:51,364 INFO - [MainThread] [airtimemediamonitorbootstrap.py : scan()] : LINE
29 - watched directories
found: {u'1': u'/srv/airtime/stor/'}
```

Use the **PageUp** and **PageDown** keys on your keyboard to navigate through the log file, or press the **q** key to quit the viewer.

To dump all log files and compress them into a tarball placed in the working directory, you could add the -d switch to the command:

```
$ sudo airtime-log -d

Creating Airtime logs tgz file at /root/logs/airtime-log-all-2011-11-29-12-43-20.tgz
```

To view just the Liquidsoap log output in real-time, you could enter the command:

```
$ sudo airtime-log -t liquidsoap

Tail liquidsoap log2011/11/29 11:44:41 [fallback_4892:3] Switch to src_4890 with forgetful
transition.
2011/11/29 11:44:41 [lang:3] /usr/lib/airtime/pypo/bin/liquidsoap_scripts/notify.sh --data='2'
--media-id=
2011/11/29 12:10:06 [server:3] New client: localhost.
2011/11/29 12:10:06 [server:3] Client localhost disconnected.
2011/11/29 12:10:06 [server:3] New client: localhost.
2011/11/29 12:10:06 [server:3] Client localhost disconnected.
2011/11/29 12:10:29 [server:3] New client: localhost.
2011/11/29 12:10:29 [server:3] Client localhost disconnected.
2011/11/29 12:10:29 [server:3] New client: localhost.
2011/11/29 12:10:29 [server:3] Client localhost disconnected.
```

Press the **Ctrl+C** keys to interrupt the real-time log output and return to the server console.

26. BACKING UP THE SERVER

The following shell commands can be used for database backup and restore on a running *PostgreSQL* server in an Airtime system.

You can dump the entire database to a zipped file with the combination of the **pg_dumpall** command and **gzip**. The **pg_dumpall** command is executed as the user *postgres*, by using the **sudo** command and the **-u** switch. It is separated from the **gzip** command with the pipe symbol.

```
$ sudo -u postgres pg_dumpall | gzip -c > airtime-backup.gz
```

This command can be automated to run on a regular basis using the standard **cron** tool on your server.

When restoring a production database on a cleanly installed Airtime system, it may be necessary to drop the empty database that was created during the new installation, by using the **dropdb** command. Again, this command is executed with **sudo** as the user *postgres*:

```
$ sudo -u postgres dropdb airtime
```

This **dropdb** command above is necessary to avoid 'already exists' errors on table creation when overwriting an empty Airtime database in the next step. These errors might prevent some data from being restored, such as user account data.

To restore, first unzip the backup file with **gunzip**, then use the **psql** command as the *postgres* user:

```
$ gunzip airtime-backup.gz
$ sudo -u postgres psql -f airtime-backup postgres
```

You should now be able to log in to the Airtime web interface in the usual way.

For safety reasons, your regular database backups should be kept in a directory which is backed up by your storage backup tool of choice; for example, the */srv/airtime/database_backups* directory. This should ensure that a storage restore can be made along with a matching and complete version of the Airtime database from the day that the storage backup was made.

STORAGE BACKUP

Backing up the Airtime database with **pg_dumpall** will not back up the Airtime media storage server, which is likely to need a great deal more backup space. Creating a compressed file from hundreds of gigabytes of storage server contents is likely to take a very long time, and may have little benefit for the amount of CPU power used, if the media files are already stored in a highly compressed format. It is also impractical to copy very large backup files across the network on a daily basis.

Instead, it is preferable to use an incremental backup technique to synchronise the production Airtime server storage with a backup server each day or night. If the backup server also contains an Airtime installation, it should be possible to switch playout to this second machine relatively quickly, in case of a hardware failure or other emergency on the production server.

A standard incremental backup tool on GNU/Linux servers is *rsync* (http://rsync.samba.org/) which can be installed using the package manager of your GNU/Linux distribution. However, incremental backup alone cannot help in the scenario where a file which later proves to be important has been deleted by an administrator. For backups that can be rolled back to restore from an earlier date than the current backup, the tool *rdiff-backup* (http://www.nongnu.org/rdiff-backup/) can be deployed.

27. UPGRADING

Airtime 2.0.0 supports upgrading from version 1.8.0 and above. If you are running a production server with a version of Airtime prior to 1.8.0, you should upgrade it to version 1.8.0 before continuing.

Before upgrading a production Airtime server, you should back up both the PostgreSQL database and the storage server used by Airtime. This is especially important if you have not already set up a regular back up routine. This extra back up is a safety measure in case of accidental data loss during the upgrade, for example due to the wrong command being entered when moving files. See the chapter *Backing up the server* in this book for details of how to perform these back ups.

If you have deployed Airtime using the method shown in the *Automated installation* chapter, you can upgrade in the same way. A new Airtime package available in the Sourcefabric repository can be installed with:

```
$ sudo apt-get update
$ sudo apt-get upgrade
```

If you have used the method shown in the *Manual installation* chapter, you should repeat the installation steps of downloading and unpacking the tarball to an installation directory, and running the **airtime-install** script. The installation script will detect an existing Airtime deployment and back up any configuration files that it finds.

After the upgrade has completed, you may need to clear your web browser's cache before logging into the new version of the Airtime administration interface. If the playout engine starts up and detects that a show should be playing at the current time, it will skip to the right point in the track and start playing.

In Airtime 1.9.0 onwards, the concept of *linked files* was replaced with the concept of *watched folders*. If you are upgrading from a version of Airtime earlier than 1.9.0 and you have previously linked files, the folders they are in will not be watched until you add them to your watched folder list. See the chapter *Manage Media Folders* for more details.

28. TROUBLESHOOTING

If your Airtime server is not working as expected, individual components of the system can be stopped, started or restarted in the server console using the **service** command:

```
$ sudo service airtime-playout        start|stop|restart|status
$ sudo service airtime-media-monitor  start|stop|restart|status
$ sudo service airtime-show-recorder  start|stop|restart|status
```

On Debian squeeze, use the command **invoke-rc.d** in place of **service** in the commands above.

The **status** option runs the **airtime-check-system** script to confirm that all of Airtime's dependencies are installed and running correctly.

Airtime stores log files under the directory path */var/log/airtime/* which can be useful for diagnosing the cause of any problems. Copies of these log files may be requested by Sourcefabric engineers while they are providing technical support for your Airtime deployment. See the chapter *The airtime-log command* for more details.

ADVANCED CONFIGURATION

29. ICECAST AND SHOUTCAST

Airtime supports direct connection to two popular streaming media servers, the open source **Icecast** (http://www.icecast.org) and the proprietary **SHOUTcast** (http://www.shoutcast.com). Apart from the software license, the main difference between these two servers is that Icecast supports simultaneous MP3 and Ogg Vorbis streaming from Airtime, whereas SHOUTcast supports MP3 but not Ogg Vorbis. The royalty-free Ogg Vorbis format has the advantage of better sound quality at lower bitrates, which has a direct impact on the amount of bandwidth that your station will require to serve the same number of listeners.

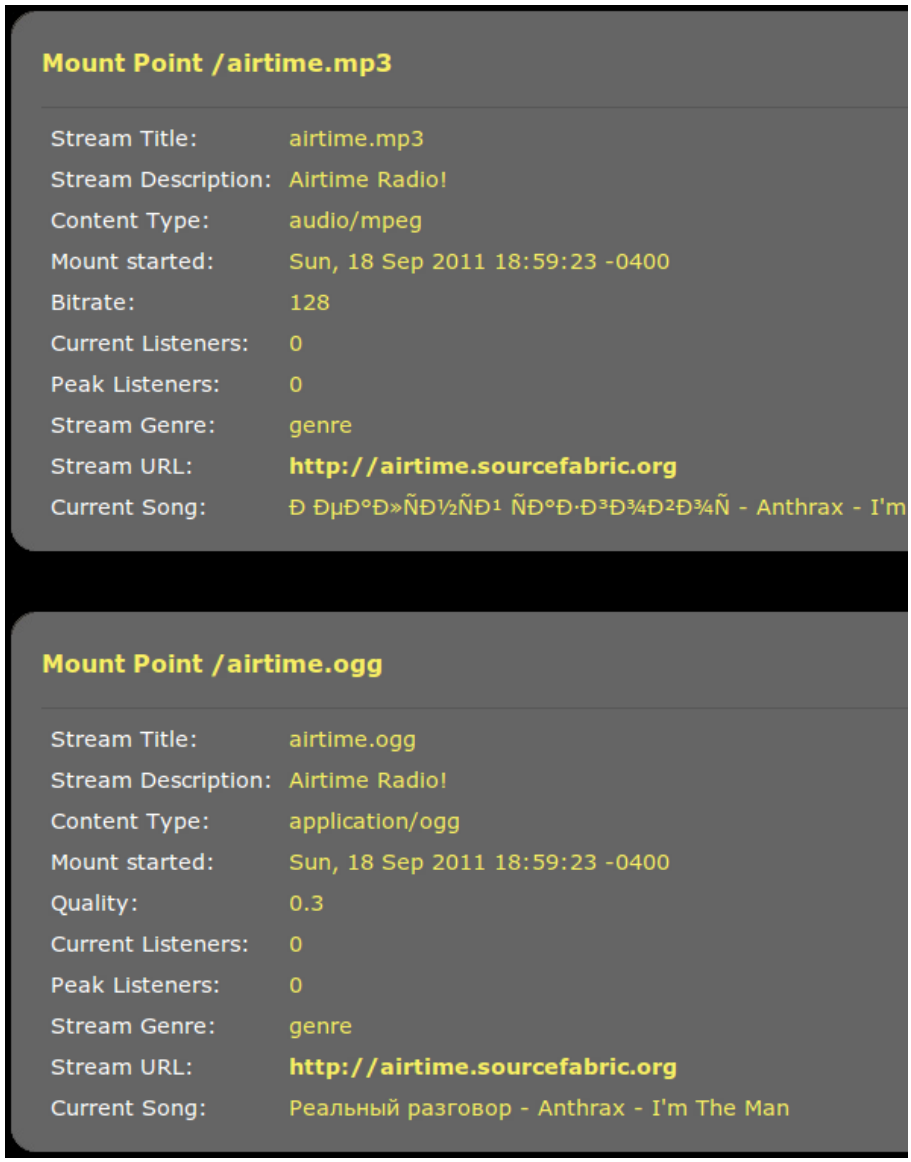
Ogg Vorbis playback has been supported natively in **Mozilla Firefox** since version 3.5, and is supported in several stand-alone media players. Combined with **jPlayer** (http://jplayer.org/), Ogg Vorbis works with all well-known platforms and web browsers. (See the chapter *Stream player for your website* on how to deliver **jPlayer** to your audience).

Streaming MP3 below a bitrate of 128kbps is not recommended for music. A 64kbps MP3 stream may be acceptable for voice broadcasts if there is a requirement for compatibility with legacy hardware playback devices which do not support Ogg Vorbis streams. Because Airtime supports simultaneous streaming in both formats, it is possible to offer one stream via your website, and another independent stream for direct connection from hardware players. You can test whether Ogg Vorbis streams sound better at low bitrates for yourself, by using the **LISTEN** button in Airtime's **Master Panel** to switch between streaming formats.

Conversely, you may have a music station which wants to stream at 160kbps or 192kbps to offer a quality advantage over stations streaming at 128kbps or less. Since both Ogg Vorbis and MP3 formats use lossy compression, listeners will only hear the benefit of higher streaming bitrates if the media files in the Airtime storage server are encoded at an equivalent bitrate, or higher.

UTF-8 METADATA IN ICECAST MP3 STREAMS

When sending metadata about your stream to an Icecast server in non-Latin alphabets, you may find that Icecast does not display the characters correctly for an MP3 stream, even though they are displayed correctly for an Ogg Vorbis stream. In the following screenshot, Russian characters are being displayed incorrectly in the Current Song field for the MP3 stream:



The solution is to specify that the metadata for the MP3 mount point you are using should be interpreted using UTF-8 encoding. You can do this by adding the following stanza to the */etc/icecast2/icecast.xml* file, where *airtime.mp3* is the name of your mount point:

```
  <mount>
       <mount-name>/airtime.mp3</mount-name>
       <charset>UTF-8</charset>
  </mount>
```

After saving the */etc/icecast2/icecast.xml* file, you should restart the Icecast server:

```
$ sudo invoke-rc.d icecast2 restart
Restarting icecast2: Starting icecast2
Detaching from the console
icecast2.
```

30. STREAM PLAYER FOR YOUR WEBSITE

If you are using Airtime for web streaming, you can embed a player applet into your website. **jPlayer** is an open source player applet (available under the GNU GPL from http://jplayer.org/) which uses the *<audio>* tag feature of HTML5 to play your streams. If the listener's browser does not support HTML5, the applet falls back to using Adobe Flash instead.

EXAMPLE CODE

You can download the example code for this chapter from:

http://en.flossmanuals.net/airtime/_booki/airtime/static/jPlayer_demo.zip

Unzip this file on your computer, and then open the file *jplayer-demo.html* in your editor:

```
$ nano jplayer-demo.html
```

We'll focus on the two areas that you need to concern yourself with. Firstly, in the *<head>* tag of the document, you'll see some code:

```
$(document).ready(function(){

 $("#jquery_jplayer_1").jPlayer({

  ready: function () {
   $(this).jPlayer("setMedia", {
     oga: "http://localhost:8000/airtime.ogg"
      }).jPlayer("play");
  },

   ended: function (event) {
    $(this).jPlayer("play");
   },

    swfPath: "js",
    supplied: "oga"

 });

});
```

This code loads jPlayer, and specifies the source of the Airtime stream. The stream setting of *http://localhost:8000/airtime.ogg* will work if you are testing jPlayer directly on the Icecast server that Airtime is connecting to. When testing on a remote server, you should change this setting to the IP address or domain name, port number and mount point of the Icecast server you are using.

As soon as jPlayer has finished loading, it will automatically begin to play the stream. The parameters *ready*, *ended*, *swfPath* and *supplied* are arguments passed to jPlayer. A full list of constructor arguments is available in the **jPlayer Developer Guide** at http://www.jplayer.org/latest/developer-guide/

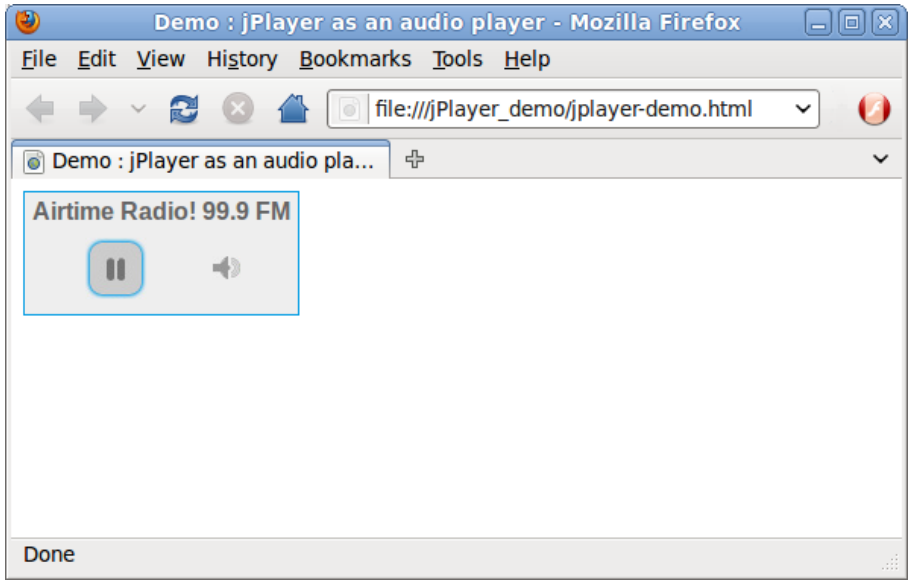
JPLAYER CONTROLS

Secondly, the *<body>* tag of the file *jplayer-demo.html* defines the controls displayed by jPlayer. jPlayer can be as simple as just one Play/Pause button, or a fully-fledged playback interface with a playlist, progress bar and volume control. In this example code, we're using one of the simpler skins available for jPlayer, *Blue Monday*.

A stop button or progress bar is not useful when we are streaming continuously from Icecast, so we can simply remove these tags from the demo code, along with the tags specifying the playback time and file duration. The simplified HTML creates a play/pause button and a mute button, and looks as follows:

```
<div id="jquery_jplayer_1" class="jp-jplayer"></div>

 <div class="jp-audio">
  <div class="jp-type-single">

   <div id="jp_interface_1" class="jp-interface">

    <div class="label">Airtime Radio! 99.9 FM</div>

     <ul class="jp-controls">
      <li><a href="#" class="jp-play" tabindex="1">play</a></li>
      <li><a href="#" class="jp-pause" tabindex="1">pause</a></li>
      <li><a href="#" class="jp-mute" tabindex="1">mute</a></li>
      <li><a href="#" class="jp-unmute" tabindex="1">unmute</a></li>
     </ul>

   </div>

   <div id="jp_playlist_1" class="jp-playlist"></div>

  </div>
 </div>
```

When you open the HTML file *jplayer-demo.html* in a web browser, your player should appear as follows:



The original version of this demo code and skin are available from http://jplayer.org/download/ for further experimentation.

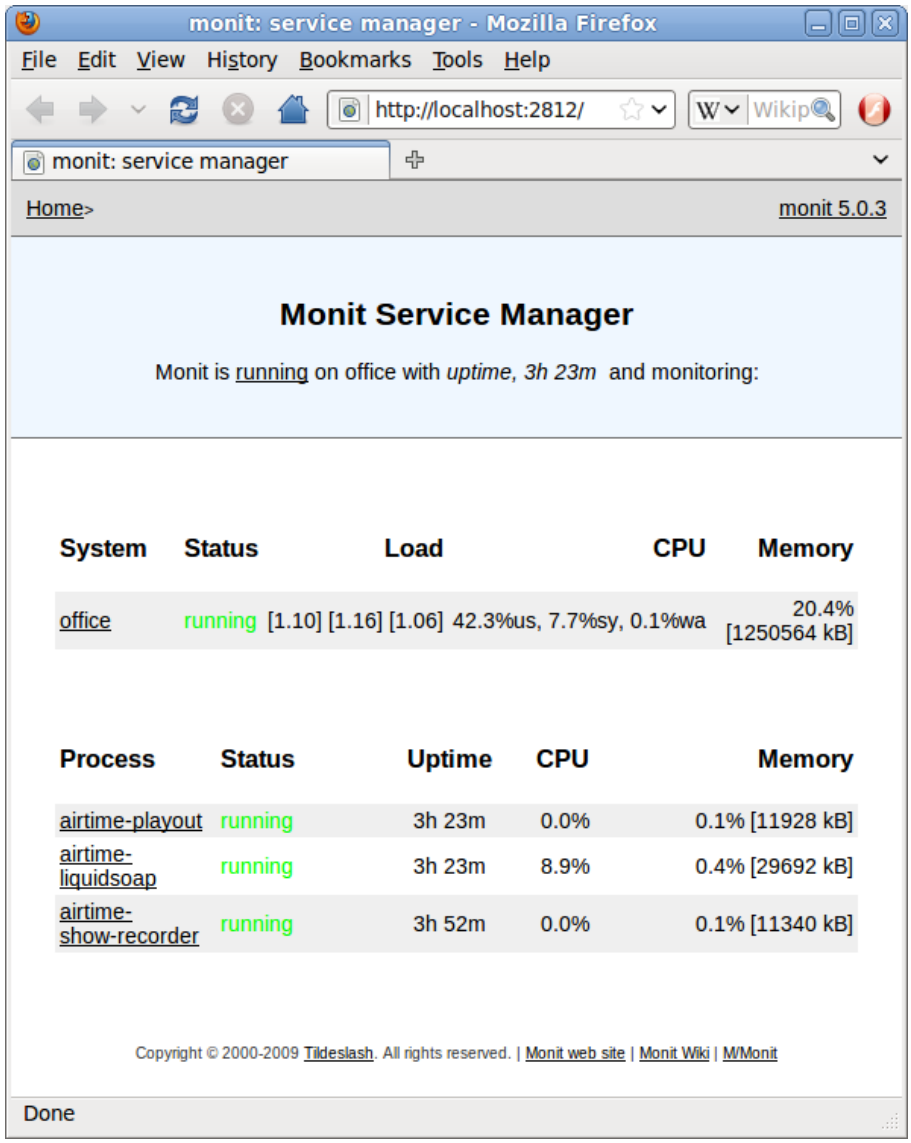
31. USING MONIT

Monit is a utility which can be used to manage and monitor processes, files, directories and filesystems on your Airtime server. It is installed by default when you install Airtime.

To view the Monit web interface, open port 2812 of your server in a web browser:

```
http://localhost:2812/
```

You will have to log in to see the Monit web interface. The default username is *admin* and the default password is *monit*.



REMOTE ACCESS

By default, Monit is configured for access from the *localhost* only. To enable remote access, you will need to edit the file */etc/monit/conf.d/airtime-monit.cfg* with **nano**:

```
$ sudo nano /etc/monit/conf.d/airtime-monit.cfg
```

Comment out the line *use address 127.0.0.1* and then add an 'allow' line with the IP address and netmask of the machine to connect to monit from, such as:

```
    allow 10.0.1.2/255.255.255.0
```

If you enable remote access, be sure to change the default admin password of *monit* in the line *allow admin:monit* to something more secure:

```
set httpd port 2812 and # use address 127.0.0.1
    allow localhost
    allow 10.0.1.2/255.255.255.0
    allow admin:pleasedonothackme
```

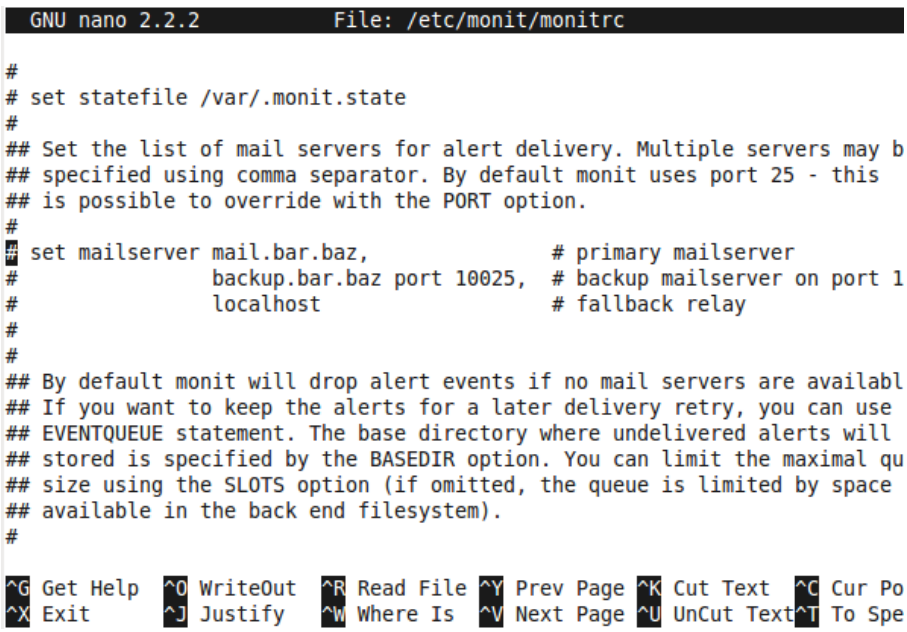
Press Ctrl+O to save the file, then Ctrl+X to exit **nano**. Then restart monit with:

```
$ sudo invoke-rc.d monit restart
```

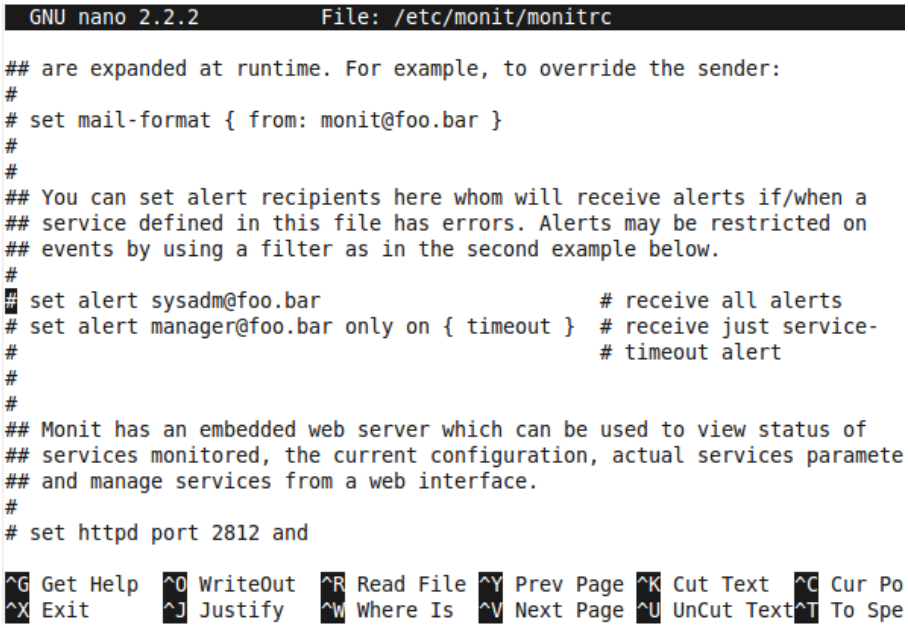
SENDING EMAIL ALERTS

To configure Monit to send email alerts, edit the file */etc/monit/monitrc* to uncomment the *set mailserver* parameter. Change this line to show the name of the SMTP server on your Airtime server's network, as provided by your system administrator.

```
$ sudo nano /etc/monit/monitrc
```



Further down the configuration file, you can set the *From:* and *To:* addresses for the alert emails using the *set mail-format* and *set alert* parameters. Then uncomment the lines required.



Press Ctrl+O to save the file, then Ctrl+X to exit **nano**. Then restart monit with:

```
$ sudo invoke-rc.d monit restart
```

More information about using Monit is available at http://mmonit.com/monit/documentation/

32. AUTOMATED FILE IMPORT

The *airtime-import* script can be combined with the standard *SFTP* (secure FTP) program and *cron* daemon on a GNU/Linux server to enable automated file import from multiple remote desktop computers. This saves time for your station staff when using distributed production methods, or content syndication.

Traditional FTP servers use plain text passwords (without encryption) and are therefore not recommended for upload accounts on Airtime servers accessible from the public Internet. SFTP is a cross-platform protocol which works with many desktop programs including **gFTP** for GNU/Linux (http://www.gftp.org/). This program can be installed on Debian or Ubuntu desktop computers with the command:

```
$ sudo apt-get install gftp
```

Other popular SFTP clients include **FileZilla** for Windows (http://filezilla-project.org/) and **Cyberduck** for Mac and Windows (http://cyberduck.ch/).

To enable SFTP uploads, first invoke the **adduser** command to create the *uploads* account on the server. For security reasons this user account is restricted to using SFTP only; it cannot be used for executing other commands in a login shell.

```
$ sudo adduser --home /srv/airtime/uploads --shell /usr/lib/sftp-server uploads
```

The server will then invite you to type in the password for the new *uploads* user, and once again for confirmation. The security of your Airtime server depends on the strength of the password that you set, so be sure to use a long and complex password with upper case, lower case and numerical characters. It is not necessary to set a full name or other details for this account.

```
Adding user `uploads' ...
Adding new group `uploads' (1003) ...
Adding new user `uploads' (1002) with group `uploads' ...
Creating home directory `/srv/airtime/uploads' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for uploads
Enter the new value, or press ENTER for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n] Y
```

Next, create a folder to contain the incoming files:

```
$ sudo mkdir /srv/airtime/uploads/incoming/
```

Then create a script to run once per hour:

```
$ sudo nano /etc/cron.hourly/airtime-upload
```

The script should import the newly uploaded files from the incoming folder specified, using the *copy* option, and then remove the original uploaded files. This step, rather than simply using the *watch* option on the */srv/airtime/uploads/incoming/* folder, ensures that the *uploads* SFTP account does not have direct write access to the Airtime storage archive. That could be a security risk if the password was compromised.

```
#!/bin/sh

# Run the import script on fresh uploads
```

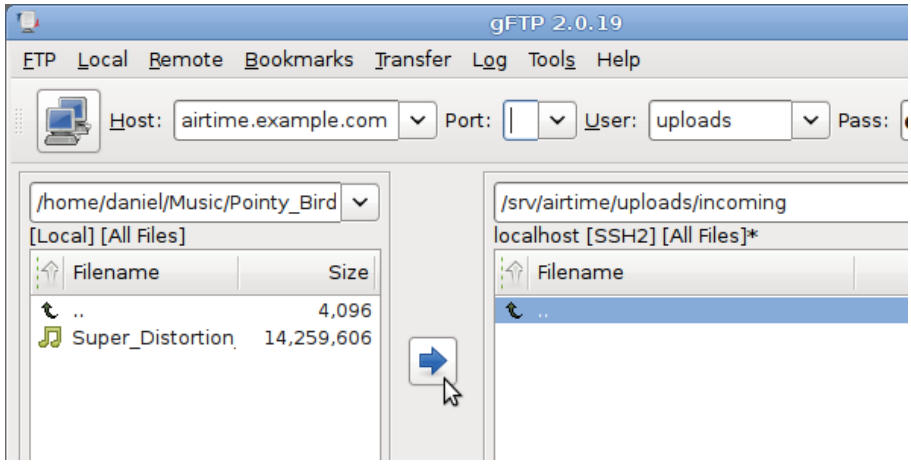
```
airtime-import copy /srv/airtime/uploads/incoming/

# Clean the incoming directory to save disk space

rm -r /srv/airtime/uploads/incoming/*.mp3
rm -r /srv/airtime/uploads/incoming/*.ogg
```

Finally, the script should be made executable so that the cron daemon can run it.

```
$ sudo chmod +x /etc/cron.hourly/airtime-upload
```

By default, Debian and Ubuntu GNU/Linux run *cron.hourly* tasks at 17 minutes past each hour. This value can be adjusted in the file */etc/crontab* on the server, if required.

Remote users should connect to the Airtime server using their client software of choice, making sure that they specify an SFTP rather than FTP connection. The remote directory for the clients to use would be */srv/airtime/uploads/incoming/* as configured above.



For additional security, you could configure your Airtime server to use an encryption key pair for the *uploads* account, instead of a password. See https://help.ubuntu.com/community/SSH/OpenSSH/Keys for details of how to do this on an Ubuntu server.

33. EXPORTING THE SCHEDULE

Airtime has a feature which enables your station's current show and schedule information to be displayed on remote websites. This feature is included in Airtime because you would not usually invite the general public to access your Airtime server directly. If you had very large numbers of people requesting data from the Airtime server at once, the burst of network traffic might overload the server, potentially disrupting your broadcasts. If carried out maliciously, this network overload is known as a *denial of service attack*.

Instead, your public-facing web server can retrieve the schedule information from Airtime. This information can then be displayed on your broadcast station or affiliate websites by a content management system, such as Sourcefabric's **Newscoop** (http://newscoop.sourcefabric.org/). It can be presented using Javascript widgets and styled with CSS, in any format that you require.

There are two kinds of information that can be retrieved remotely from Airtime; the metadata for the current show plus the following show (live-info), or the schedule for the current week (week-info). This metadata includes show names, times, descriptions and individual show URLs on your public website. That way, the audience for your station can click through from the schedule information to find out more about a particular show, or download a previous show recording that you might have made available.

If your Airtime server was accessible at *http://airtime.example.com* the live show information could be retrieved by your web server using this URL:

```
http://airtime.example.com/api/live-info/?callback
```

The comma-separated text metadata returned to your web server might be something like this:


```
({
"env":"development",
"schedulerTime":"2011-05-09 15:01:18",
"currentShow":[{"start_timestamp":"2011-05-09 16:00:00",
"end_timestamp":"2011-05-09 17:00:00",
"name":"Funk Show",
"id":"8",
"instance_id":"8",
"record":"0",
"url":"http:\/\/funk.example.com\/"}],
"nextShow":[{"id":"9","starts":"2011-05-09 17:00:00",
"ends":"2011-05-09 18:00:00",
"show_id":"9",
"record":"0",
"rebroadcast":"0",
"instance_id":null,
"file_id":null,
"soundcloud_id":null,
"time_filled":null,
"name":"Dance show",
"url":"http:\/\/dance.example.com",
"genre":"Dance",
"description":"Techno, techno, techno, techno!",
"color":"000000",
"background_color":"ffea00",
"start_timestamp":"2011-05-09 17:00:00",
"end_timestamp":"2011-05-09 18:00:00"}],
"timezone":"BST",
"timezoneOffset":"3600"
})
```

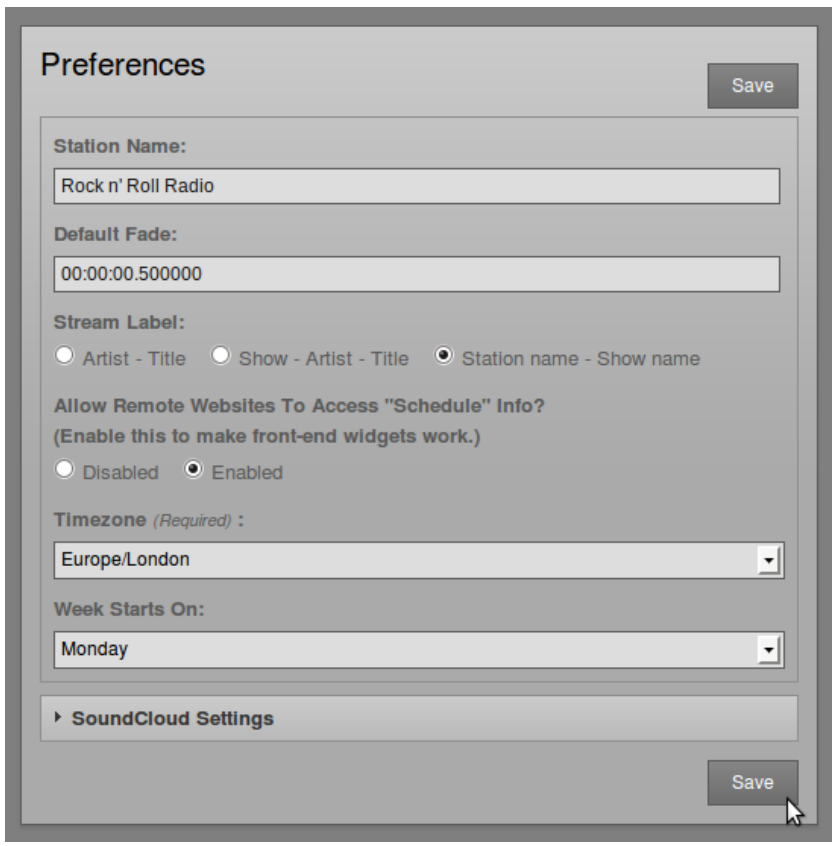

The information for the current week's schedule could be retrieved using the URL:

```
http://airtime.example.com/api/week-info/?callback
```

In this case, the metadata returned would be in a different format from the above example, something like the following. To keep the example short, this particular schedule export only contains four shows on a Monday. A full weekly schedule export would contain a great deal more text.


```
({
"sunday":[],
"monday":[
{"show_starts":"2011-05-09 14:25:00",
"show_ends":"2011-05-09 14:35:00",
"show_name":"Elvis Show",
"url":"http:\/\/elvis.example.com\/"},
{"show_starts":"2011-05-09 14:50:00",
"show_ends":"2011-05-09 14:55:00",
"show_name":News",
"url":"http:\/\/news.example.com\/"},
{"show_starts":"2011-05-09 16:00:00",
"show_ends":"2011-05-09 17:00:00",
"show_name":"Funk Show",
"url":"http:\/\/funk.example.com\/"},
{"show_starts":"2011-05-09 17:00:00",
"show_ends":"2011-05-09 18:00:00",
"show_name":"Dance show",
"url":"http:\/\/dance.example.com"}
],
"tuesday":[],
"wednesday":[],
"thursday":[],
"friday":[],
"saturday":[]
})
```


If you see the message *You are not allowed to access this resource* when attempting to display schedule information in your web browser, log in to the Airtime administration interface, click Configure in the main menu, then Preferences. Set **Allow Remote Websites To Access "Schedule" Info?** to **Enabled**, click the **Submit** button, then refresh the browser window opened on the schedule export URL.



CACHING SCHEDULE INFORMATION

If the Airtime server is behind a firewall, or you want to protect the Airtime server from large numbers of schedule requests, you may wish to cache the schedule information on a public-facing or intermediate server. You can then create a firewall rule that only allows the schedule server to connect to the Airtime server, in addition to any remote users of the Airtime web interface.

Your system administrator can set up schedule caching on a standard Apache and PHP enabled web server with the *curl* program installed, using the following steps:

1. Create a bash script on the schedule server that polls the Airtime server, and writes the metadata returned into a pair of temporary files:

```
$ sudo nano /usr/local/bin/airtime-schedule.sh
```

The content of this file should be like the following script, replacing *air1.example.com* with the name of your Airtime server:

```
#!/bin/sh

curl -s "http://air1.example.com/api/live-info/?callback=***" > /tmp/live-info

curl -s "http://air1.example.com/api/week-info/?callback=***" > /tmp/week-info
```

2. Make the bash script executable:

```
$ sudo chmod +x /usr/local/bin/airtime-schedule.sh
```

3. Create an Apache VirtualHost configuration for the schedule server:

```
$ sudo nano /etc/apache2/sites-available/schedule
```

containing a definition like the following, replacing *schedule.example.com* with the name of your schedule server:

```
<VirtualHost *:80>
   ServerName schedule.example.com
   DocumentRoot /var/www/schedule/
</VirtualHost>
```

4. In the schedule server's DocumentRoot folder, create the folders *api/live-info/* and *api/week-info/*

```
$ sudo mkdir -p /var/www/schedule/api/live-info/
$ sudo mkdir -p /var/www/schedule/api/week-info/
```

5. Create an index.php file in the *api/live-info/* folder:

```
$ sudo nano /var/www/schedule/api/live-info/index.php
```

containing the following code:

```
<?php
$filename = '/tmp/live-info';  // define here the path and name of uploaded live-info file

header('Content-Type: text/javascript');
header("Expires: Thu, 01 Jan 1970 00:00:00 GMT");
header("Cache-Control: no-store, no-cache, must-revalidate");

$callback = empty($_GET['callback']) ? null : $_GET['callback'];
$content = file_get_contents($filename);
$content = str_replace('***', $callback, $content);
echo $content;
?>
```

6. Create an index.php file in the *api/week-info/* folder:

```
$ sudo nano /var/www/schedule/api/week-info/index.php
```

containing the following code:

```
<?php
$filename = '/tmp/week-info';  // define here the path and name of uploaded week-info file

header('Content-Type: text/javascript');
header("Expires: Thu, 01 Jan 1970 00:00:00 GMT");
header("Cache-Control: no-store, no-cache, must-revalidate");

$callback = empty($_GET['callback']) ? null : $_GET['callback'];
$content = file_get_contents($filename);
$content = str_replace('***', $callback, $content);
echo $content;
?>
```

7. Enable the new configuration and reload the Apache web server:

```
$ sudo a2ensite schedule
$ sudo /etc/init.d/apache2 reload
```

8. Create a cron job to run the bash script each minute:

```
$ sudo nano /etc/cron.d/airtime-schedule
```

containing the line:

```
* * * * * www-data /usr/local/bin/airtime-schedule.sh
```

The schedule server will now be serving the same show information as the Airtime server, with a cache lifetime of one minute. You can adjust the cache lifetime by altering the frequency of the cron job that polls the Airtime server.

WEBSITE WIDGETS

Example HTML, Javascript and CSS code for your public website are provided in the *widgets* folder of the Airtime installation tarball. For the widgets to work on a typical web server, links to the Javascript and CSS code have to be included in the HTML page <head> element, like the following example:

```
<head>
 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title>Airtime widgets</title>
   <script src="js/jquery-1.6.1.min.js" type="text/javascript">
    </script>
   <script src="js/jquery-ui-1.8.10.custom.min.js" type="text/javascript">
    </script>
   <script src="js/jquery.showinfo.js" type="text/javascript">
    </script>
  <link href="css/airtime-widgets.css" rel="stylesheet" type="text/css" />
</head>
```

A full example is shown in the *widgets/sample_page.html* file in the Airtime installation tarball.

The following code is for a small *airtimeLiveInfo* widget that displays information about the current show (show time elapsed, and show time remaining), as well as some information about the next show (start time and end time). In this example, the label text for *onAirNow* is translated into French for local language support:

```
<script>
    $(document).ready(function() {
        $("#headerLiveHolder").airtimeLiveInfo({
            sourceDomain: "http://schedule.example.com/",
            text: {onAirNow:"Sur Les Antennes", offline:"Offline", current:"Current",
next:"Next"},
            updatePeriod: 20 //seconds
        });
    });
</script>
```

On the public website, this widget can be made to look like the following screenshot:

The CSS properties *color:* and *text-transform:uppercase* have been used to style the *onAirNow* label. There is a full example CSS file *widgets/css/airtime-widgets.css* in the Airtime installation tarball.

The next widget *airtimeShowSchedule* is medium sized, and displays the upcoming show schedule for that day.

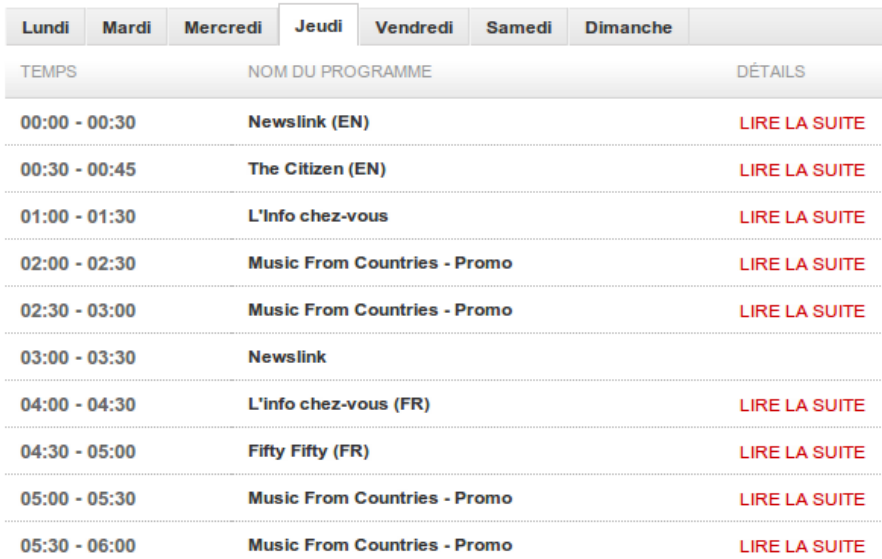
```
<script>
    $(document).ready(function() {
        $("#onAirToday").airtimeShowSchedule({
            sourceDomain: "http://schedule.example.com/",
            text: {onAirToday:"On air today"},
            updatePeriod: 5 //seconds
        });
    });
</script>
```

The widget code above can be styled to look like this screenshot:

Finally, the following code creates a large widget *airtimeWeekSchedule* that enables site visitors to browse through the show schedule for that week. In this example, all widget labels have been translated into French:

```
<script>
    $(document).ready(function() {
        $("#scheduleTabs").airtimeWeekSchedule({
            sourceDomain:"http://schedule.example.com/",
            dowText:{monday:"Lundi", tuesday:"Mardi", wednesday:"Mercredi", thursday:"Jeudi",
friday:"Vendredi", saturday:"Samedi", sunday:"Dimanche"},
            miscText:{time:"Temps", programName:"Nom du Programme", details:"Détails",
readMore:"Lire La Suite"},
            updatePeriod: 600 //seconds
        });
    });
</script>
```

Using the code above and CSS, the first six hours of the schedule each day can be styled to look like this:



Lundi | Mardi | Mercredi | Jeudi | Vendredi | Samedi | Dimanche

TEMPS	NOM DU PROGRAMME	DÉTAILS
00:00 - 00:30	Newslink (EN)	LIRE LA SUITE
00:30 - 00:45	The Citizen (EN)	LIRE LA SUITE
01:00 - 01:30	L'Info chez-vous	LIRE LA SUITE
02:00 - 02:30	Music From Countries - Promo	LIRE LA SUITE
02:30 - 03:00	Music From Countries - Promo	LIRE LA SUITE
03:00 - 03:30	Newslink	
04:00 - 04:30	L'info chez-vous (FR)	LIRE LA SUITE
04:30 - 05:00	Fifty Fifty (FR)	LIRE LA SUITE
05:00 - 05:30	Music From Countries - Promo	LIRE LA SUITE
05:30 - 06:00	Music From Countries - Promo	LIRE LA SUITE

The value of **sourceDomain** in the code examples above should match the URL that you wish to serve schedule information to the public from. If you have used the *Caching schedule information* method detailed above, this would be the URL of your schedule server, not the Airtime server directly.

34. INTERFACE CUSTOMIZATION

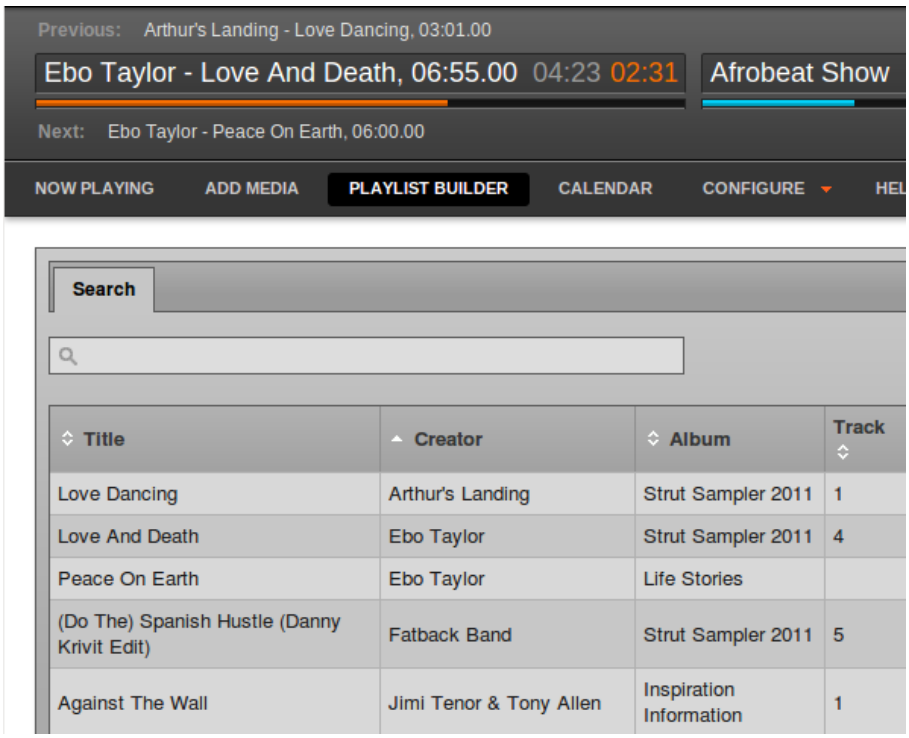
The Airtime administration interface, as a web application, is fully customizable using the same methods that you might use to update a website. For instance, you may wish to increase the font sizes or change the colours in the Airtime interface to better suit staff users with impaired vision. To do this, open one of the CSS files in the */public/css/* directory under the Airtime *DocumentRoot* directory in an editor such as **nano**:

```
$ sudo nano /usr/share/airtime/public/css/styles.css
```

To change the background colour of the administration interface from dark gray to white, the *background:* property of the body tag could be changed to *#ffffff* as follows:

```
body {
      font-size: 62.5%;
      font-family:Arial, Helvetica, sans-serif;
      background: #ffffff;
      margin: 0;
      padding: 0;
}
```

Save the file with Ctrl+O, then refresh your browser to see the change to the interface background colour.



Any custom changes that you make to the administration interface should be backed up before upgrading Airtime to a newer version, otherwise they could be overwritten. If you have made improvements that you think might be useful to other Airtime users, please contact Sourcefabric and tell us about them.

MODIFYING THE ICECAST INTERFACE

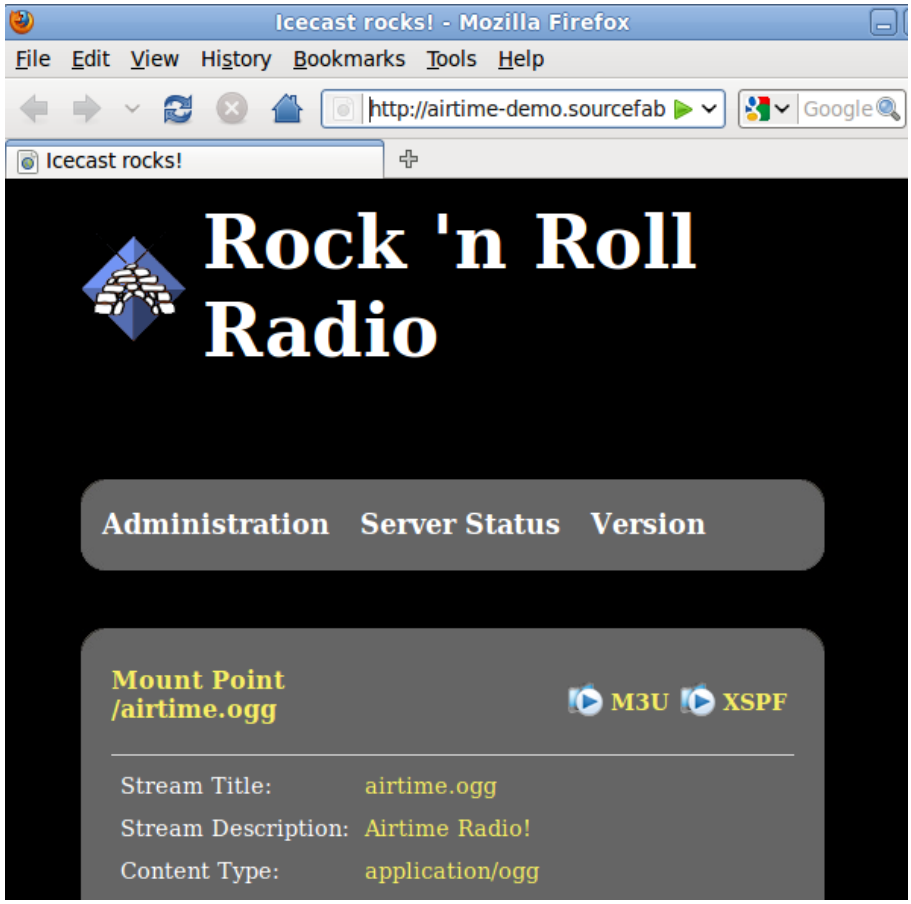
If you have installed Icecast, in the directory */etc/icecast2/web/* you will find several XSLT and other files which are used to generate the Icecast web interface. If you are familiar with HTML you should be able to modify these pages, as they are well commented. You do have to be careful with syntax, because something as simple as a missing bracket can cause the Icecast web interface to break down.

For example, you could change the *status.xsl* page:

```
$ sudo nano /etc/icecast2/web/status.xsl
```

Modifying the *status.xsl* page is a good place to start, because this is the default page that site visitors see when they browse port 8000 on your Icecast server. The most obvious change to make in the XSLT pages is the content of the *<title>* and *<h2>* tags, to announce the name of your station. You can also modify the *style.css* file in this directory to change colour and layout options.

After saving the file with Ctrl+O, refresh your web browser, and the new look should now be visible.



When you're happy with the way the web interface looks and the sound quality you are streaming, you can uncomment the *<directory>* section in the */etc/icecast2/icecast.xml* file to have your new station automatically listed on the Icecast directory website http://dir.xiph.org which could help you pick up a few more listeners. You can also put a link to the Icecast status page at port 8000 on your station's homepage, to help integrate the two sites.

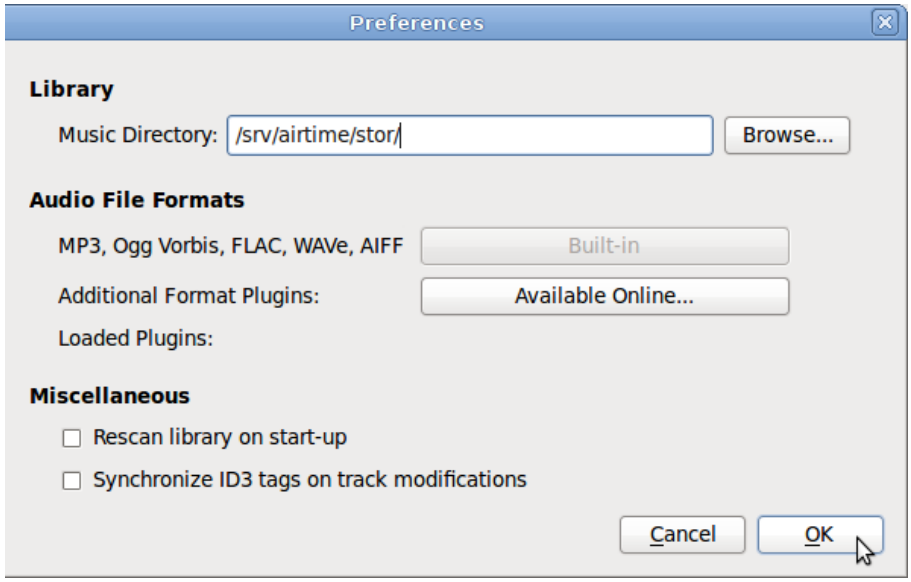
35. INTEGRATION WITH MIXXX

Mixxx is a cross-platform Open Source application for DJs, available from http://www.mixxx.org/

Installed on a desktop or laptop computer, Mixxx complements your Airtime server to provide a complete system for both live and scheduled broadcasting. Although Mixxx has many features designed for dance music DJs that require beat matching and pitch independent time stretching, the program can be used for any kind of manually triggered broadcast playout, including live speech shows such as news or current affairs.

Mixxx supports a wide variety of popular hardware control surfaces, which can be connected to your computer using a USB cable. A control surface might replace or augment an analogue mixer in your studio, depending on your live mixing and playout requirements.

If you make the Airtime server's storage directory */srv/airtime/stor/* accessible to a desktop machine as a read-only location, Mixxx will accept that location as its default music library when starting up for the first time. (This location can also be configured after installation by clicking Options, then Preferences, then Library in the main Mixxx menu).



You may need to adjust file and directory permissions so that the storage directory has read access from the desktop user account. Enabling write access directly to the storage server is not recommended, as this would allow desktop users to delete files which might be needed for playout later.

If the filesystem path has been configured correctly, the metadata for the files in the Airtime storage server will be displayed in the main window of the Mixxx interface. Individual files from the Airtime storage server can then be added to either of Mixxx's live players with a right-click on the filename, or by using the appropriate hardware buttons on a control surface. Therefore it is possible to manage the station's storage archive remotely and collaboratively through Airtime, while using Mixxx as the live playout client in multiple, remote studios.

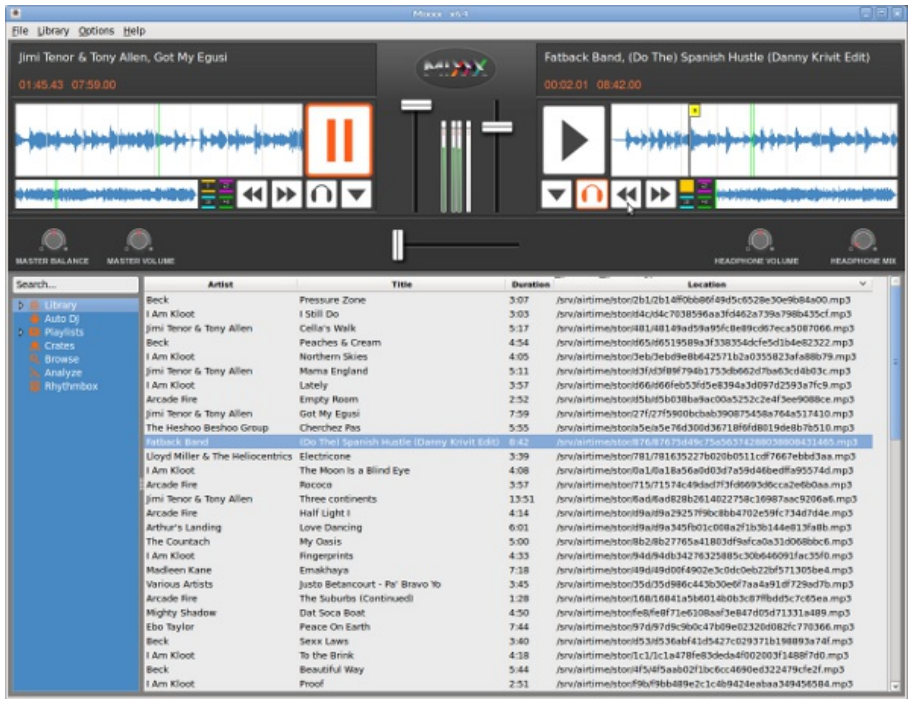
The Airtime storage archive can be exported like any other file server share. The method that you implement would depend on the operating system of your desktop client machines, and whether they were on the same local network as the Airtime server, or remote. For performance and redundancy reasons it is advisable to cache files required for a particular show on the client machine where Mixxx is installed. For example, for a GNU/Linux client machine, a nightly **rsync** download of new media in the archive would guard against network problems at playout time potentially disrupting a broadcast at a remote studio.

Mixxx users can also record a show, encode it, and then upload it through the Airtime web interface on a local or remote server for collaborative or user-generated broadcasts. In addition, Mixxx 1.9.0 or later includes a live streaming client which, like Airtime, is compatible with the **Icecast** media server.

AIRTIME SKIN FOR MIXXX

An Airtime themed skin for Mixxx, designed with broadcast users in mind, is available for download from https://sourceforge.net/projects/airtime/files/

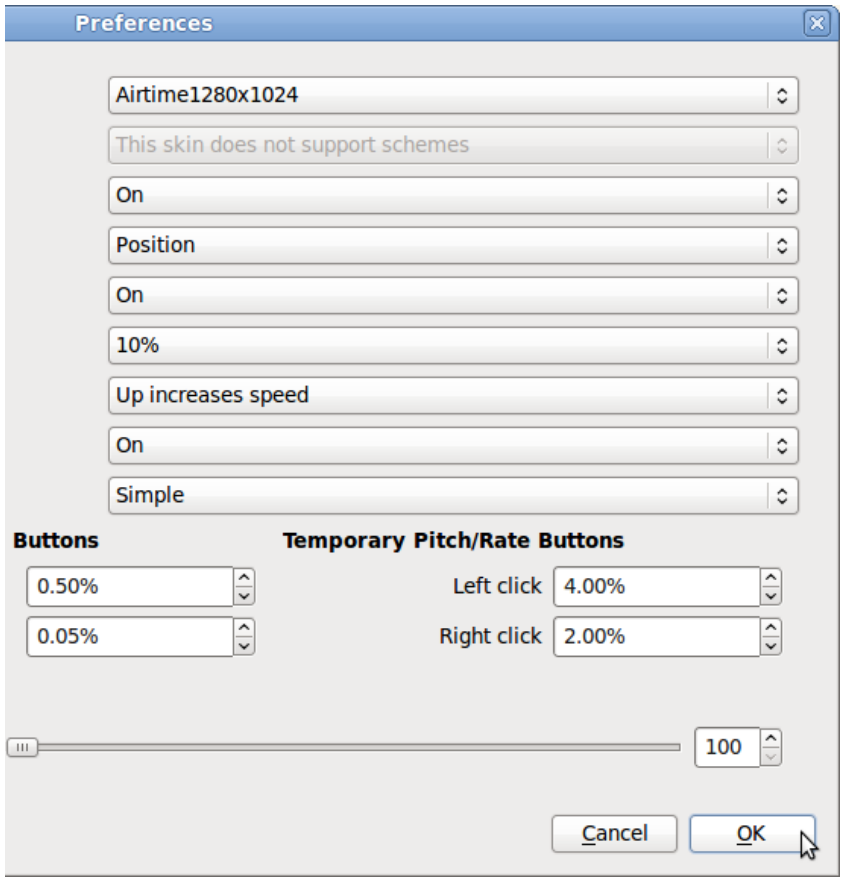
This skin is a simplified interface for live broadcasting which does away with pitch/tempo control, EQ, flange effect, looping and other features required by beat-matching dance music DJs. Instead, the emphasis is on a clear and uncluttered interface which does not require large mouse movements to operate the most important controls.



After downloading, extract the zip file and copy it to the *skins* directory on the computer where Mixxx is installed. For example, on Debian or Ubuntu:

```
$ unzip Airtime1280x1024_skin_for_Mixxx.zip
$ sudo cp -r Airtime1280x1024 /usr/share/mixxx/skins/
```

Then, start Mixxx and select the Airtime skin by clicking *Options*, *Preferences*, then *Interface* in the Mixxx main menu.


Preferences
Airtime1280x1024
This skin does not support schemes
On
Position
On
10%
Up increases speed
On
Simple
Buttons
0.50%
0.05%
Temporary Pitch/Rate Buttons
Left click
4.00%
Right click
2.00%
100
Cancel
OK

36. STREAM HANDOVER

In a typical configuration, the live output from the broadcast studio and the scheduled output from the storage archive are mixed together before being sent further along the broadcast chain, such as to a transmitter or streaming media server on the Internet.

If your Airtime server is hosted in a remote data centre, you may not have the option to handover the streaming media source from a live to scheduled show manually, because you have no physical access to connect a broadcast mixer to the server. Alternatively, you may have an Airtime server at the studio, connected to your main mixer, but wish stream handovers to take place automatically at the correct times. Disconnecting the stream and beginning another is less than ideal, because the audience's media players will also be disconnected when that happens.

The **Icecast** server has a *fallback-mount* feature which can be used to move clients (media players used by listeners or viewers) from one source to another, as new sources become available. This makes it possible to handover from Airtime scheduled output to a live show from another source, and handover to Airtime again once the live show has ended.

To enable fallback mounts, edit the main Icecast configuration file to define the mount points you will use, and the relationship between them.

```
$ sudo nano /etc/icecast2/icecast.xml
```

The example *<mount>* section provided in the *icecast.xml* file is commented out by default. Before or after the commented section, add three mount point definitions. The default mount point used by Airtime is */airtime.ogg* which is shown in the */etc/airtime/liquidsoap.cfg* file. You must also define a mount point for the live source (called */live.ogg* in this example) and a mount point for the public to connect to (called */stream.ogg* in this example).

```
<mount>
     <mount-name>/airtime.ogg</mount-name>
     <hidden>0</hidden>
</mount>

<mount>
     <mount-name>/live.ogg</mount-name>
     <fallback-mount>/airtime.ogg</fallback-mount>
     <fallback-override>1</fallback-override>
     <hidden>0</hidden>
</mount>

<mount>
     <mount-name>/stream.ogg</mount-name>
     <fallback-mount>/live.ogg</fallback-mount>
     <fallback-override>1</fallback-override>
     <hidden>0</hidden>
</mount>
```

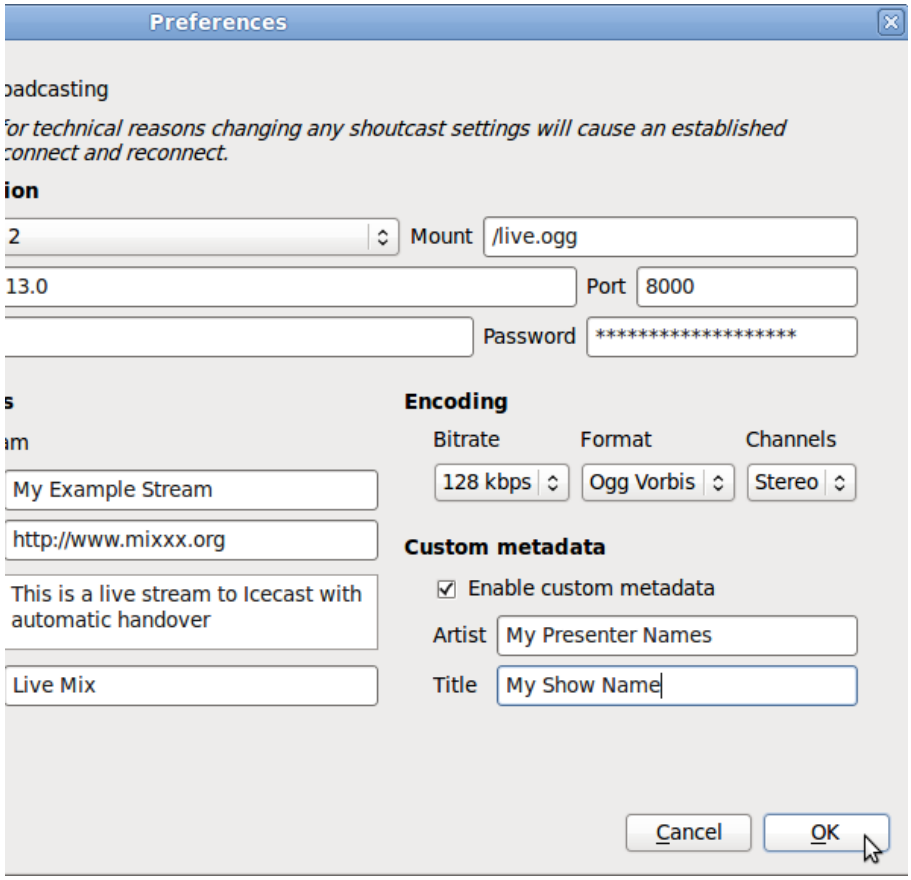
These mount point definitions mean that a client connecting to a URL such as *http://icecast.example.com:8000/stream.ogg* will first fall back to the */live.ogg* mount point if it is available. If not, the client will fall back in turn to the */airtime.ogg* mount point for scheduled show playout.

Setting the value of *<fallback-override>* to 1 (enabled) means that when the */live.ogg* mount point becomes available again, the client will be re-connected to it. If you wish to hide the */airtime.ogg* and */live.ogg* mount points from the public Icecast web interface, set the value of *<hidden>* in each of these definitions to 1.

LIVE SOURCE CONFIGURATION

To integrate the live streaming source with Airtime, connect the live source to the Icecast server with the same parameters defined in the */etc/airtime/liquidsoap.cfg* file, except for the mount point. This should be the live mount point you have defined in the */etc/icecast2/icecast.xml* file, such as */live.ogg* in the example above.

To configure **Mixxx** for streaming to Icecast, click *Options*, *Preferences*, then *Live Broadcasting*. For server *Type*, select the default of *Icecast 2* when streaming to Debian or Ubuntu servers, as this is the version of Icecast supplied with those GNU/Linux distributions.



By default, Icecast streams are buffered to guard against network problems, which causes latency for remote listeners. When monitoring the stream from a remote location, you may have to begin the live stream a few seconds before the previous stream ends to enable a smooth transition.

APPENDIX

37. EXPERT INSTALL

These quick install steps are suitable for experienced GNU/Linux system administrators who have already followed the steps shown in the chapter *Preparing the server* earlier in this book. For a more detailed explanation of the steps below, please read the chapter *Automated installation*.

1. Edit the repositories file for your server:

```
$ sudo nano /etc/apt/sources.list
```

For Ubuntu Lucid [or Maverick, Natty, Oneiric] servers, use the Sourcefabric repository:

```
deb http://apt.sourcefabric.org/ lucid main
```

and make sure you have enabled the multiverse repository for MP3 encoding support:

```
deb http://archive.ubuntu.com/ubuntu/ lucid multiverse
```

For Debian Squeeze servers, use the Sourcefabric repository:

```
deb http://apt.sourcefabric.org/ squeeze main
```

and enable the backports repository for MP3 encoding support:

```
deb http://backports.debian.org/debian-backports squeeze-backports main
```

2. Install the Sourcefabric package signing key:

```
$ sudo apt-get update
$ sudo apt-get install sourcefabric-keyring
```

3. Install the database management system:

```
$ sudo apt-get install postgresql
```

4. Install the streaming media server (optional):

```
$ sudo apt-get install icecast2
```

5. Remove PulseAudio:

```
$ sudo apt-get purge pulseaudio
```

6. Install Airtime:

```
$ sudo apt-get install airtime
```

Refer to the *Configuration* chapter for configuration options. Now you should be able to log in to the Airtime administration interface, as shown in the *Getting started* chapter.

38. HD AUDIO MODELS

This listing is provided to help ensure that the correct model parameter is passed to the ALSA kernel module for an Intel HDA soundcard, if one is fitted to your Airtime server. See the chapter *Preparing the server* in this book for more details.

```
  Model name Description
  ----------    -----------
ALC880
======
  3stack 3-jack in back and a headphone out
  3stack-digout 3-jack in back, a HP out and a SPDIF out
  5stack 5-jack in back, 2-jack in front
  5stack-digout 5-jack in back, 2-jack in front, a SPDIF out
  6stack 6-jack in back, 2-jack in front
  6stack-digout 6-jack with a SPDIF out
  w810  3-jack
  z71v  3-jack (HP shared SPDIF)
  asus  3-jack (ASUS Mobo)
  asus-w1v ASUS W1V
  asus-dig ASUS with SPDIF out
  asus-dig2 ASUS with SPDIF out (using GPIO2)
  uniwill 3-jack
  fujitsu Fujitsu Laptops (Pi1536)
  F1734  2-jack
  lg  LG laptop (m1 express dual)
  lg-lw  LG LW20/LW25 laptop
  tcl  TCL S700
  clevo  Clevo laptops (m520G, m665n)
  medion Medion Rim 2150
  test  for testing/debugging purpose, almost all controls can be
  adjusted.  Appearing only when compiled with
  $CONFIG_SND_DEBUG=y
  auto  auto-config reading BIOS (default)

ALC260
======
  hp  HP machines
  hp-3013 HP machines (3013-variant)
  hp-dc7600 HP DC7600
  fujitsu Fujitsu S7020
  acer  Acer TravelMate
  will  Will laptops (PB V7900)
  replacer Replacer 672V
  favorit100 Maxdata Favorit 100XS
  basic  fixed pin assignment (old default model)
  test  for testing/debugging purpose, almost all controls can
  adjusted.  Appearing only when compiled with
  $CONFIG_SND_DEBUG=y
  auto  auto-config reading BIOS (default)

ALC262
======
  fujitsu Fujitsu Laptop
  hp-bpc HP xw4400/6400/8400/9400 laptops
  hp-bpc-d7000 HP BPC D7000
  hp-tc-t5735 HP Thin Client T5735
  hp-rp5700 HP RP5700
  benq  Benq ED8
  benq-t31 Benq T31
  hippo  Hippo (ATI) with jack detection, Sony UX-90s
  hippo_1 Hippo (Benq) with jack detection
  sony-assamd Sony ASSAMD
  toshiba-s06 Toshiba S06
  toshiba-rx1 Toshiba RX1
  tyan  Tyan Thunder n6650W (S2915-E)
  ultra  Samsung Q1 Ultra Vista model
  lenovo-3000 Lenovo 3000 y410
  nec  NEC Versa S9100
  basic  fixed pin assignment w/o SPDIF
  auto  auto-config reading BIOS (default)

ALC267/268
==========
  quanta-il1 Quanta IL1 mini-notebook
```

```
  3stack 3-stack model
  toshiba Toshiba A205
  acer  Acer laptops
  acer-dmic Acer laptops with digital-mic
  acer-aspire Acer Aspire One
  dell  Dell OEM laptops (Vostro 1200)
  zepto  Zepto laptops
  test  for testing/debugging purpose, almost all controls can
  adjusted.  Appearing only when compiled with
  $CONFIG_SND_DEBUG=y
  auto  auto-config reading BIOS (default)

ALC269
======
  basic  Basic preset
  quanta Quanta FL1
  laptop-amic Laptops with analog-mic input
  laptop-dmic Laptops with digital-mic input
  fujitsu FSC Amilo
  lifebook Fujitsu Lifebook S6420
  auto  auto-config reading BIOS (default)

ALC662/663/272
==============
  3stack-dig 3-stack (2-channel) with SPDIF
  3stack-6ch  3-stack (6-channel)
  3stack-6ch-dig 3-stack (6-channel) with SPDIF
  5stack-dig  5-stack with SPDIF
  lenovo-101e  Lenovo laptop
  eeepc-p701 ASUS Eeepc P701
  eeepc-ep20 ASUS Eeepc EP20
  ecs  ECS/Foxconn mobo
  m51va  ASUS M51VA
  g71v  ASUS G71V
  h13  ASUS H13
  g50v  ASUS G50V
  asus-mode1 ASUS
  asus-mode2 ASUS
  asus-mode3 ASUS
  asus-mode4 ASUS
  asus-mode5 ASUS
  asus-mode6 ASUS
  asus-mode7 ASUS
  asus-mode8 ASUS
  dell  Dell with ALC272
  dell-zm1 Dell ZM1 with ALC272
  samsung-nc10 Samsung NC10 mini notebook
  auto  auto-config reading BIOS (default)

ALC680
======
  base  Base model (ASUS NX90)
  auto  auto-config reading BIOS (default)

ALC882/883/885/888/889
======================
  3stack-dig 3-jack with SPDIF I/O
  6stack-dig 6-jack digital with SPDIF I/O
  arima  Arima W820Di1
  targa  Targa T8, MSI-1049 T8
  asus-a7j ASUS A7J
  asus-a7m ASUS A7M
  macpro MacPro support
  mb5  Macbook 5,1
  macmini3 Macmini 3,1
  mba21  Macbook Air 2,1
  mbp3  Macbook Pro rev3
  imac24 iMac 24'' with jack detection
  imac91 iMac 9,1
  w2jc  ASUS W2JC
  3stack-2ch-dig 3-jack with SPDIF I/O (ALC883)
  alc883-6stack-dig 6-jack digital with SPDIF I/O (ALC883)
  3stack-6ch    3-jack 6-channel
  3stack-6ch-dig 3-jack 6-channel with SPDIF I/O
  6stack-dig-demo  6-jack digital for Intel demo board
  acer  Acer laptops (Travelmate 3012WTMi, Aspire 5600, etc)
  acer-aspire Acer Aspire 9810
  acer-aspire-4930g Acer Aspire 4930G
  acer-aspire-6530g Acer Aspire 6530G
  acer-aspire-7730g Acer Aspire 7730G
  acer-aspire-8930g Acer Aspire 8930G
  medion Medion Laptops
```

```
  targa-dig Targa/MSI
  targa-2ch-dig Targa/MSI with 2-channel
  targa-8ch-dig Targa/MSI with 8-channel (MSI GX620)
  laptop-eapd   3-jack with SPDIF I/O and EAPD (Clevo M540JE, M550JE)
  lenovo-101e Lenovo 101E
  lenovo-nb0763 Lenovo NB0763
  lenovo-ms7195-dig Lenovo MS7195
  lenovo-sky Lenovo Sky
  haier-w66 Haier W66
  3stack-hp HP machines with 3stack (Lucknow, Samba boards)
  6stack-dell Dell machines with 6stack (Inspiron 530)
  mitac  Mitac 8252D
  clevo-m540r Clevo M540R (6ch + digital)
  clevo-m720 Clevo M720 laptop series
  fujitsu-pi2515 Fujitsu AMILO Pi2515
  fujitsu-xa3530 Fujitsu AMILO XA3530
  3stack-6ch-intel Intel DG33* boards
  intel-alc889a Intel IbexPeak with ALC889A
  intel-x58 Intel DX58 with ALC889
  asus-p5q ASUS P5Q-EM boards
  mb31  MacBook 3,1
  sony-vaio-tt  Sony VAIO TT
  auto  auto-config reading BIOS (default)

ALC861/660
==========
  3stack 3-jack
  3stack-dig 3-jack with SPDIF I/O
  6stack-dig 6-jack with SPDIF I/O
  3stack-660 3-jack (for ALC660)
  uniwill-m31 Uniwill M31 laptop
  toshiba Toshiba laptop support
  asus  Asus laptop support
  asus-laptop ASUS F2/F3 laptops
  auto  auto-config reading BIOS (default)

ALC861VD/660VD
==============
  3stack 3-jack
  3stack-dig 3-jack with SPDIF OUT
  6stack-dig 6-jack with SPDIF OUT
  3stack-660 3-jack (for ALC660VD)
  3stack-660-digout 3-jack with SPDIF OUT (for ALC660VD)
  lenovo Lenovo 3000 C200
  dallas Dallas laptops
  hp  HP TX1000
  asus-v1s ASUS V1Sn
  auto  auto-config reading BIOS (default)

CMI9880
=======
  minimal 3-jack in back
  min_fp 3-jack in back, 2-jack in front
  full  6-jack in back, 2-jack in front
  full_dig 6-jack in back, 2-jack in front, SPDIF I/O
  allout 5-jack in back, 2-jack in front, SPDIF out
  auto  auto-config reading BIOS (default)

AD1882 / AD1882A
================
  3stack 3-stack mode (default)
  6stack 6-stack mode

AD1884A / AD1883 / AD1984A / AD1984B
====================================
  desktop 3-stack desktop (default)
  laptop laptop with HP jack sensing
  mobile mobile devices with HP jack sensing
  thinkpad Lenovo Thinkpad X300
  touchsmart HP Touchsmart

AD1884
======
  N/A

AD1981
======
  basic  3-jack (default)
  hp  HP nx6320
  thinkpad Lenovo Thinkpad T60/X60/Z60
  toshiba Toshiba U205
```

```
AD1983
======
  N/A

AD1984
======
  basic  default configuration
  thinkpad Lenovo Thinkpad T61/X61
  dell_desktop Dell T3400

AD1986A
=======
  6stack 6-jack, separate surrounds (default)
  3stack 3-stack, shared surrounds
  laptop 2-channel only (FSC V2060, Samsung M50)
  laptop-eapd 2-channel with EAPD (ASUS A6J)
  laptop-automute 2-channel with EAPD and HP-automute (Lenovo N100)
  ultra  2-channel with EAPD (Samsung Ultra tablet PC)
  samsung 2-channel with EAPD (Samsung R65)
  samsung-p50 2-channel with HP-automute (Samsung P50)

AD1988/AD1988B/AD1989A/AD1989B
==============================
  6stack 6-jack
  6stack-dig ditto with SPDIF
  3stack 3-jack
  3stack-dig ditto with SPDIF
  laptop 3-jack with hp-jack automute
  laptop-dig ditto with SPDIF
  auto  auto-config reading BIOS (default)

Conexant 5045
=============
  laptop-hpsense    Laptop with HP sense (old model laptop)
  laptop-micsense   Laptop with Mic sense (old model fujitsu)
  laptop-hpmicsense Laptop with HP and Mic senses
  benq  Benq R55E
  laptop-hp530 HP 530 laptop
  test  for testing/debugging purpose, almost all controls
  can be adjusted.  Appearing only when compiled with
  $CONFIG_SND_DEBUG=y

Conexant 5047
=============
  laptop Basic Laptop config
  laptop-hp Laptop config for some HP models (subdevice 30A5)
  laptop-eapd Laptop config with EAPD support
  test  for testing/debugging purpose, almost all controls
  can be adjusted.  Appearing only when compiled with
  $CONFIG_SND_DEBUG=y

Conexant 5051
=============
  laptop Basic Laptop config (default)
  hp  HP Spartan laptop
  hp-dv6736 HP dv6736
  hp-f700 HP Compaq Presario F700
  ideapad Lenovo IdeaPad laptop
  lenovo-x200 Lenovo X200 laptop
  toshiba Toshiba Satellite M300

Conexant 5066
=============
  laptop Basic Laptop config (default)
  hp-laptop HP laptops, e g G60
  asus  Asus K52JU, Lenovo G560
  dell-laptop Dell laptops
  dell-vostro Dell Vostro
  olpc-xo-1_5 OLPC XO 1.5
  ideapad       Lenovo IdeaPad U150
  thinkpad Lenovo Thinkpad

STAC9200
========
  ref  Reference board
  oqo  OQO Model 2
  dell-d21 Dell (unknown)
  dell-d22 Dell (unknown)
  dell-d23 Dell (unknown)
  dell-m21 Dell Inspiron 630m, Dell Inspiron 640m
  dell-m22 Dell Latitude D620, Dell Latitude D820
  dell-m23 Dell XPS M1710, Dell Precision M90
```

```
  dell-m24 Dell Latitude 120L
  dell-m25 Dell Inspiron E1505n
  dell-m26 Dell Inspiron 1501
  dell-m27 Dell Inspiron E1705/9400
  gateway-m4 Gateway laptops with EAPD control
  gateway-m4-2 Gateway laptops with EAPD control
  panasonic Panasonic CF-74
  auto  BIOS setup (default)

STAC9205/9254
=============
  ref  Reference board
  dell-m42 Dell (unknown)
  dell-m43 Dell Precision
  dell-m44 Dell Inspiron
  eapd  Keep EAPD on (e.g. Gateway T1616)
  auto  BIOS setup (default)

STAC9220/9221
=============
  ref  Reference board
  3stack D945 3stack
  5stack D945 5stack + SPDIF
  intel-mac-v1 Intel Mac Type 1
  intel-mac-v2 Intel Mac Type 2
  intel-mac-v3 Intel Mac Type 3
  intel-mac-v4 Intel Mac Type 4
  intel-mac-v5 Intel Mac Type 5
  intel-mac-auto Intel Mac (detect type according to subsystem id)
  macmini Intel Mac Mini (equivalent with type 3)
  macbook Intel Mac Book (eq. type 5)
  macbook-pro-v1 Intel Mac Book Pro 1st generation (eq. type 3)
  macbook-pro Intel Mac Book Pro 2nd generation (eq. type 3)
  imac-intel Intel iMac (eq. type 2)
  imac-intel-20 Intel iMac (newer version) (eq. type 3)
  ecs202 ECS/PC chips
  dell-d81 Dell (unknown)
  dell-d82 Dell (unknown)
  dell-m81 Dell (unknown)
  dell-m82 Dell XPS M1210
  auto  BIOS setup (default)

STAC9202/9250/9251
==================
  ref  Reference board, base config
  m1  Some Gateway MX series laptops (NX560XL)
  m1-2  Some Gateway MX series laptops (MX6453)
  m2  Some Gateway MX series laptops (M255)
  m2-2  Some Gateway MX series laptops
  m3  Some Gateway MX series laptops
  m5  Some Gateway MX series laptops (MP6954)
  m6  Some Gateway NX series laptops
  auto  BIOS setup (default)

STAC9227/9228/9229/927x
=======================
  ref  Reference board
  ref-no-jd Reference board without HP/Mic jack detection
  3stack D965 3stack
  5stack D965 5stack + SPDIF
  5stack-no-fp D965 5stack without front panel
  dell-3stack Dell Dimension E520
  dell-bios Fixes with Dell BIOS setup
  volknob Fixes with volume-knob widget 0x24
  auto  BIOS setup (default)

STAC92HD71B*
============
  ref  Reference board
  dell-m4-1 Dell desktops
  dell-m4-2 Dell desktops
  dell-m4-3 Dell desktops
  hp-m4  HP mini 1000
  hp-dv5 HP dv series
  hp-hdx HP HDX series
  hp-dv4-1222nr HP dv4-1222nr (with LED support)
  auto  BIOS setup (default)

STAC92HD73*
===========
  ref  Reference board
  no-jd  BIOS setup but without jack-detection
```

```
  intel  Intel DG45* mobos
  dell-m6-amic Dell desktops/laptops with analog mics
  dell-m6-dmic Dell desktops/laptops with digital mics
  dell-m6 Dell desktops/laptops with both type of mics
  dell-eq Dell desktops/laptops
  alienware Alienware M17x
  auto  BIOS setup (default)

STAC92HD83*
===========
  ref  Reference board
  mic-ref Reference board with power management for ports
  dell-s14 Dell laptop
  hp  HP laptops with (inverted) mute-LED
  hp-dv7-4000 HP dv-7 4000
  auto  BIOS setup (default)

STAC9872
========
  vaio  VAIO laptop without SPDIF
  auto  BIOS setup (default)

Cirrus Logic CS4206/4207
========================
  mbp55  MacBook Pro 5,5
  imac27 IMac 27 Inch
  auto  BIOS setup (default)

VIA VT17xx/VT18xx/VT20xx
========================
  auto  BIOS setup (default)
```

39. ABOUT THIS MANUAL

GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
- b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to “keep intact all notices”.
- c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
- d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

- a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
- b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
- c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
- d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
- e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
- c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
- e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's “contributor version”.

A contributor's “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS