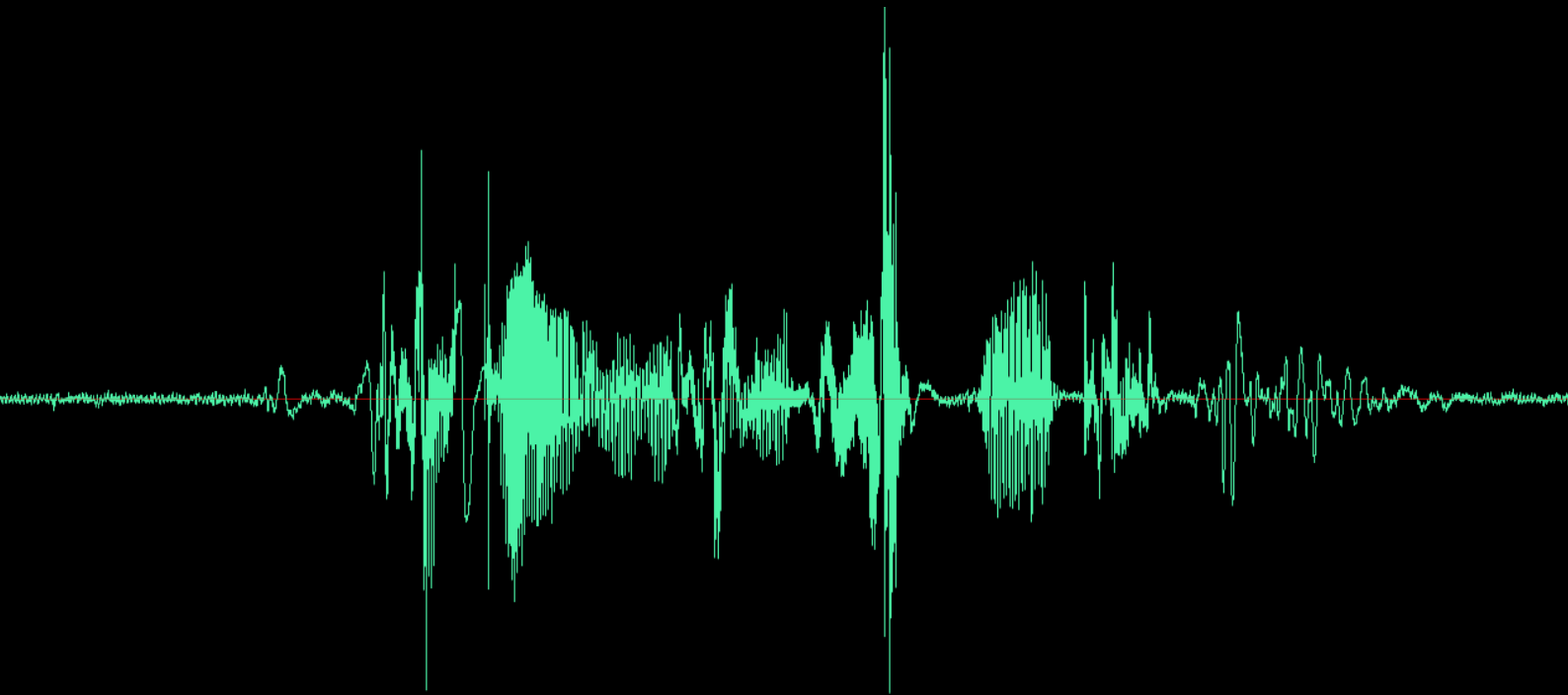


podcast

Play that funky music



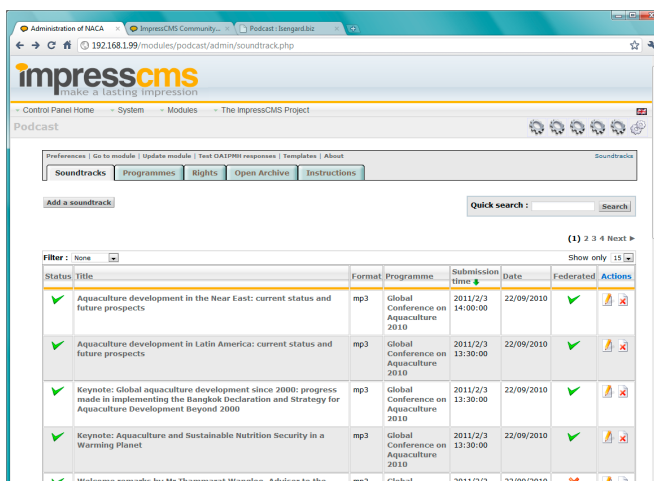
Contents

What does the Podcast module do?	2
Installation and set up	2
• Optional: Installing an inline audio/video player (JW Player)	2
User-side views	2
Podcast administration	3
• Programmes	3
• Soundtracks	3
• Adding more formats (mimetypes)	4
• Rights	4
• Archive	4
Configuring Podcast preferences	5
Open Archives functionality	5
• Enabling your archive	6
• Testing Podcast's OAIPMH responses	6
Other stuff	6
• How streaming works in Podcast	6
• Technical considerations for streaming	6
• Audio capture / editing software	7
• Podcasting clients	7
Need help?	7
Want some custom modifications made to Podcast?	7
Annex 1: Programme submission form	8
Annex 2: Soundtrack submission form	8
Annex 3: Open Archive submission form	9

What does the Podcast module do?

Podcast is an ImpressCMS module designed principally for publishing audio podcasts via the web and RSS feeds (although it also handles video equally well). You can also use it to publish music albums, talks from conferences or other collections of sound. Key features include:

- Display, download and stream individual soundtracks or entire programmes/albums.
- Organise soundtracks into programmes (eg. albums or collections).
- Individual RSS feeds with media enclosures for each programme (W3C validated), enabling automatic download of soundtracks by podcasting clients.
- Configurable user-side display - you can turn various metadata fields on or off directly in the module preferences, without having to modify templates.
- Configurable extended/compact views to suit display of programmes/albums.
- Configurable rights management system with per-track license control.
- Dynamic image resizing - you can rebuild all your cover/album art simply by changing the display size settings in module preferences.
- Transparent participation in distributed digital library systems via an implementation of the Open Archives Initiative Protocol for Metadata Harvesting.
- It's a native ImpressCMS (IPF) module.



The administrative interface is native IPF. You can sort the list by clicking on the column headings or with the filters in the header row. You can also toggle soundtracks on or off and enable/disable federation by clicking on the status/federation icons.

Installation and set up

Installation follows standard ImpressCMS procedure:

1. Extract the podcast directory from the compressed archive.
2. Upload it into the /modules directory of your ImpressCMS install.
3. Go to System => Modules in the administration section of your website, and click on the action button to install the Podcast module.

Podcast is immediately ready for use and you can start uploading soundtracks right away. However, you probably want to set up some programmes/albums to suit your needs first.

Optional: Installing an inline audio/video player (JW Player)

If you would like people to be able to listen/view audio and video files inline within your website (in addition to being able to download and stream them), then you need to install an inline media player. Why? Because in this day and age as incredible as it may seem there is still, somehow, NO UNIVERSALLY ACCEPTED STANDARD VIDEO FORMAT. The closest thing is MP4/H.264, which is supported by all contemporary browsers, except for Internet Explorer 7 and 8 (support was finally added in version 9, which shipped with Windows 7). To provide wide compatibility you are basically forced to offer video in two formats simultaneously: MP4/H.264 and Flash.

Podcast includes support for JW Player 6, which can handle this for you automatically, serving the most appropriate file type for the client browser. If you encode your videos in MP4/H.264, JW Player will serve these files with either HTML 5 controls or in a Flash wrapper, depending on the capabilities of the client. Practically speaking, you can just produce your videos in MP4 and forget about it.

There are several editions of JW Player and one of them is free for non-commercial use. Installation is simple:

1. Download JW Player from: <http://www.longtailvideo.com/jw-player/> and follow the supplied installation instructions (drop the folder in your web root).
2. In the Podcast module preferences, set 'Enable JW Player for inline audio/video' to 'yes', and set the height/width preferences to the same dimensions you plan to encode your videos. The default setting is 480 x 360 pixels.
3. The JW Player website will also offer to give you a key for free analytics (you need to create an account there to get it). You can enter your key in the corresponding preference field here. Bear in mind potential privacy concerns (analytics = tracking).

That's it. You should now see player controls below the description when you view an individual audio or video track. Use is described elsewhere.

User side views

Podcast has four main user-side display pages (screenshots below), listed as submenu items in the main menu block:

- **New:** Displays the latest soundtracks in chronological order, across all programmes, in compact or extended view.
- **Programmes:** Displays a list of programmes, default sort order (title or date) can be configured in module preferences.
- **Soundtracks:** Displays a table of soundtracks, which can be sorted according to various column fields.
- **Archive:** Displays summary information about Podcast's Open Archives Initiative functionality, including its base URL.

Podcast administration

There are four kinds of object you will work with in Podcast:

- **Programmes** are collections of soundtracks. Use them to create podcast shows, albums, or other collections of sound.
- **Soundtracks** are individual audio files. Every soundtrack must be assigned to a programme.
- **Rights** are intellectual property licenses, such as 'copyright', 'public domain' and the various Creative Commons licenses. Every soundtrack must be assigned a license.
- **Open Archive** is a special object that most people won't need. It is used to enable cross-site federation and search of your soundtrack records, using the Open Archives Initiative Protocol for Metadata Harvesting (OAIPMH).

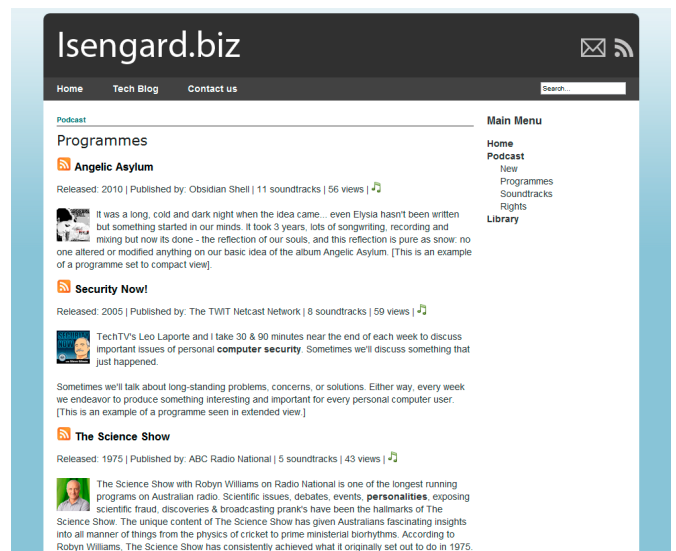
Programmes

You can create programmes (eg. podcasts, albums or collections) in the Programmes tab of the administration interface. Simply click the 'add programme' button and fill in the form. When you add soundtracks, you can choose which programme you want to assign them to. You can sort your programmes by title, publisher and submission time by clicking on the column headings.

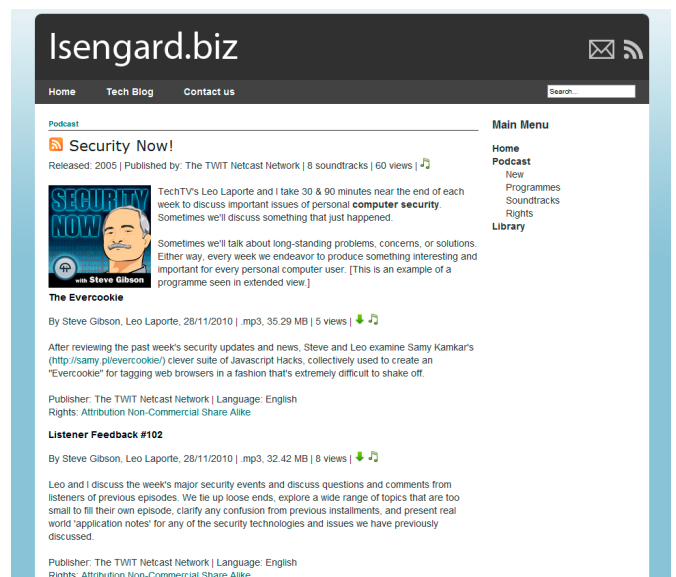
I strongly encourage you to fill in all fields in the submission form, this will make the user-side display of the module nicer to look at. If you decide you don't want to display a particular field, you can turn it off in the module preferences (see the preferences section of this manual for more information).

In particular, try to upload a logo or cover art for every programme. These are important navigational cues in the Podcast interface. Images are resized dynamically, so don't worry about submitting an image a bit bigger than what you think you need. Podcast will rebuild screenshots and thumbnails for you automatically, according to the sizes you specify in the module preferences, which you can change at any time.

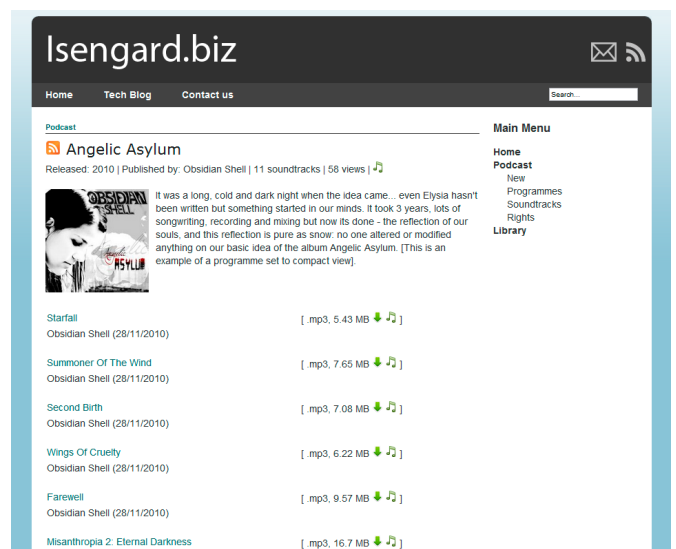
It is best to upload an image that is a bit bigger than you think you will actually need, in case you decide you want to increase the image display size preference later on. See Annex 1 for a list of the form fields and their recommended usage.



The 'programmes' index page displays the available podcasts/albums. The music icon triggers streaming of all soundtracks in that programme.



An example of a programme in 'extended view'. This is best for podcast shows where each episode has a description.



An example of a programme in 'compact view'. This format is best for albums and other collections where individual soundtracks don't have descriptions.

Soundtracks

You can add soundtracks in the Soundtracks tab of the administration interface, just click the 'add' button and fill in the form, which is largely self-explanatory. You can toggle soundtracks online or offline, and choose to federate them or not by clicking on the status and federation icons.

You can filter the list of soundtracks using the dropdown box at the head of the table (by online status, federation or license), and sort soundtracks by title, format, programme, submission time and publication date by clicking on the column headings.

I strongly recommend you fill in all fields. Some are compulsory, including title, author, URL, format and file size, as these are required in order for the module's RSS feeds to meet W3C validation requirements.

The primary media URL field is critical. This must be a **direct link** to the audio or video file. If you get this wrong the download, streaming and inline viewing functions won't work, so make sure you check each one carefully! Note that the file doesn't have to be on your server, you can republish audio or video files hosted elsewhere because it's just a link. It is also critically important to select the right format in order for the RSS feeds to work correctly.

There is also a secondary media URL field, which is optional. You can use this to serve an alternate media file which will be displayed inline via the optional JW Player, if installed. You can use to provide (say) a lower quality audio file for people with slow connections, or perhaps a higher quality one for people with fast connections. You can also use it to provide a video recording of an event while serving the audio version through your podcast feeds, or vice versa! Note that this secondary media file is only available through the inline player, it will not be included in your feeds, streams or download links. However, if you do not provide a secondary media URL then JW Player will also use your primary for inline media display. Note that you can disable JW Player in the module preferences.

If a podcast has multiple authors or contributors, separate them with a pipe character, eg. 'John Smith|Jane Doe'. Again, this is important for user-side display and RSS feed validation.

File size must be entered in bytes, Podcast will convert it to more readable forms (KB, MB) automatically for you. Make sure you get it right, some podcasting clients stop playing after they reach the reported file size - even if the soundtrack isn't finished!

You can control which of these fields are displayed on the user side of the module in the preferences section. See Annex 2 for a summary of the soundtrack submission form fields and recommended usage.

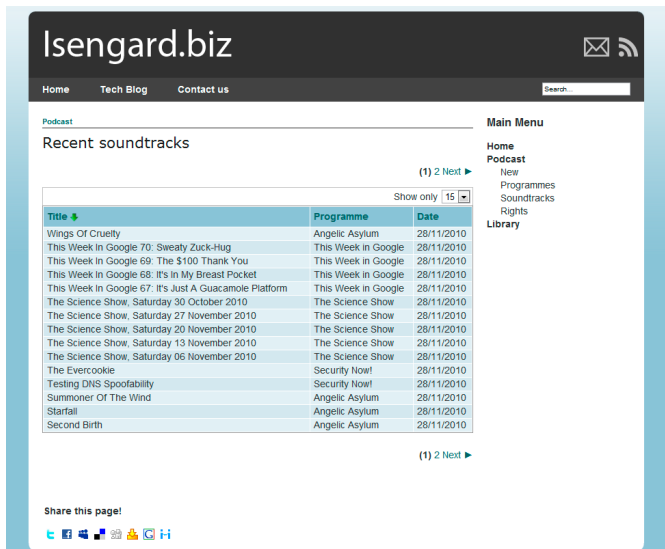
Adding more formats (mimetypes)

Podcast uses the ImpressCMS mimetype manager to control what file formats it supports. Several audio formats are enabled by default (mp3, wav and wma).

To add (or remove) formats from the list in the soundtrack submission form, go to System => Administration => Mimetypes. If the mimetype you want to add/remove is in the list, click the edit link and add/remove Podcast from the list of approved modules. If the mimetype is missing (for example, ogg), then you click the 'create a mimetype' button to add it. You must use the official mimetype declaration, which you can get from IANA: <http://www.iana.org/assignments/media-types/index.html>

You may also need to edit a file in your ImpressCMS install, open the file listed below, if the format is not already in the list, then add it to the array here as well (a pre-patched version of this file including .mp4 and .ogg is available in the extras folder of the Podcast module, which you can use to overwrite the original).

/libraries/icms/Utils.php



The 'soundtracks' page lists all tracks on the system. You can sort the list by clicking on the column headings.



View of an individual soundtrack. The green icons trigger the track to download and stream, respectively. The cover carries a link back to the programme this track is associated with.

Again, you must use the official IANA mimetype. As an example, for MP4 (which is not available by default) you would add a line to the end of the list like this:

```
"mp4" => "video/mp4"
```

This file is used to output the correct mimetype for media enclosures in the RSS feed. Basically if you don't follow the IANA spec, your RSS feeds will break and nobody will be able to get your podcasts.

Rights

Podcast ships with the most common intellectual property licenses pre-installed, ie. copyright, the Creative Commons licenses, and Public Domain. You can choose what license you want to assign to individual soundtracks as you create them. You can also add more licenses if you wish or edit the existing ones via the 'Rights' tab in the administration interface. Rights are available as selection options in the soundtrack submission form.

The default license is 'copyright', in keeping with the Berne Convention, which is the basis for copyright law in most countries. If you don't want to display the 'rights' field on the user of the module, you can turn it off in the preferences section.

Archive

Archive is a special object that most people won't need. It is used to enable federation and cross-site search of your soundtrack records, using the Open Archives Initiative Protocol for Metadata Harvesting (OAIPMH). Basically, it controls responses to incoming OAIPMH requests. See the section on 'Open Archives functionality' for a full explanation.

If you don't want to use the OAIPMH functionality then you can just ignore it. In that case, I recommend that you disable Open Archive functionality in the module preferences.

If you do want to use OAIPMH, then create an archive object in the Open Archive tab. You can only create one. Many of the fields are automatically filled for you. Some fields have been made read-only because they are fixed values that must not be changed. See Annex 3 for a list of submission form fields and their recommended usage.

Configuring Podcast preferences

Podcast is highly configurable and offers a lot of control over how the user side of the module is displayed. You can turn most metadata fields on or off for both programmes and soundtracks, to suit your own taste.

For example, if you don't want to display the rights field for soundtracks, or would like to hide the view counters, you can just flip a toggle in the preferences to hide them. Have a look through the preferences and familiarise yourself with the options.

There are also a number of preferences that govern other issues, and these are summarised in table 1 on the following page.

Open Archives functionality

Podcast provides a minimal implementation of the Open Archives Initiative Protocol for Metadata Harvesting (OAIPMH). This protocol allows specialised search engine spiders (Z39.50 metadata harvesters) to request information about your site and export its records via an XML stream. The records collected by harvesters are incorporated into external search databases.

This is basically a way of building distributed digital library systems. For example, a university library system can collect publication records from dozens of other libraries and add them to its own search catalogue. When someone searches their library catalogue they simultaneously search the records of all libraries that have been scanned - but the records are stored locally.

If they find something of interest, a link will send them to the original source library to access the underlying resource (in this case, audio files). Think 'cross-site search'. Your publications become more visible and accessible and your site gets extra traffic and backlinks. Everyone wins.

At present, it is mainly universities, research institutes and high-end digital library systems like ePrints that can sharing content through OAIPMH, but now you can too. If you have soundtracks worth sharing and wish to participate in such a network (European Schoolnet is a good example of a massive international distributed library system), contact a network administrator and offer them the 'base URL' for your podcast module. They will then be able to configure their metadata harvester to query your Podcast installation periodically to import any new soundtracks you have published and chosen to federate.

Table 1: Configuring Podcast preferences

Preference	Usage
Start page	Select which page you want Podcast to open on by default (choose from the programmes page, soundtracks, rights, or new items).
Enable Open Archive functionality?	Kill switch for OAIPMH. Podcast will not respond to OAIPMH requests if this switch is set to 'no'.
Federate soundtracks by default?	Controls the default federation setting for new soundtracks, you can override it when filling in the submission form, its just a convenience to save you time.
Default language	Controls the default language setting in the soundtrack submission form. Again, just a convenience, you can override it.
Screenshot width (in pixels)	Programme logos/album covers are shown at two different sizes in the module, this is the larger one used when a programme or soundtrack is displayed in single view mode. If you change this value, screenshots will be rebuilt at the new size automatically.
Thumbnail width (in pixels)	This is the smaller size of logo/cover, used when programmes (or new soundtracks) are displayed in a list. If you change this value, thumbnails will be automatically rebuilt at the new size automatically.
Display new podcasts page in compact view?	You can choose to have the new soundtracks page display items in compact form (a list of titles), or extended view (with descriptions).
Number of soundtracks to display on the 'new' page and in RSS feeds	Controls how many soundtracks are displayed on the new soundtracks page and included in RSS feeds. If there are more soundtracks than this, pagination controls will be inserted on the page.
Number of soundtracks to display per page	Controls the number of soundtracks displayed internally within a programme, if there are more pagination controls will be inserted.
Number of programmes to display per page	Controls the number of programmes listed on the programmes page; if there are more pagination controls will be inserted.
Sort programmes by:	Title, or by submission date (ascending or descending).
Display breadcrumb navigation?	Turns the module breadcrumb on or off.

Enabling your archive

To enable the OAIPMH functionality of Podcast, you must do three things:

1. Create an Archive object via the Open Archive tab of the administration interface. You can configure the name and various other attributes of your archive there (note that some fields are read-only).
2. Enable Open Archive functionality in the module preferences (it is enabled by default).
3. When you create soundtracks you want to share, set the 'Federation' field as 'yes'. This makes the soundtrack accessible via the OAIPMH web service (anything you mark as 'no' will not be returned in OAIPMH responses). Soundtrack 'status' must also be set as online. Soundtrack Federation and Status are both enabled by default.

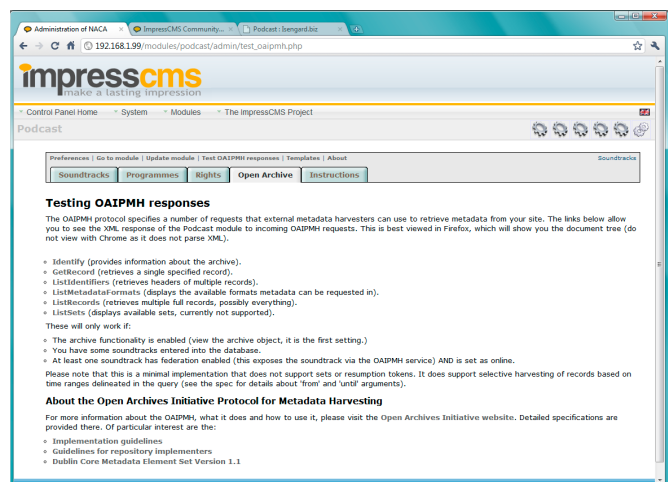
Note that there is a user-side Open Archive display page, which provides essential details about your archive including the base URL. It is found at:

http://www.yoursite.com/modules/podcast/oaipmh_target.php

Testing Podcast's OAIPMH responses

Note that there is a 'Test OAIPMH responses' link above the tabs in the administration interface. This displays some links that will trigger responses against the standard OAIPMH vocabulary so you can check if it is working and see what the response looks like. The standard queries are:

- Identify (provides basic information about the archive).
- GetRecord (retrieves a single specified record specified by its OAI identifier).
- ListIdentifiers (retrieves headers of multiple records, supports optional selective harvesting by time range).
- ListMetadataFormats (shows the available formats that metadata can be requested in, currently only unqualified Dublin Core in this implementation).
- ListRecords (retrieves multiple full records, supports optional selective harvesting by time range).



The 'test OAIPMH responses' page carries links that you can use to see the XML output generated by Podcast in response to incoming OAIPMH requests.

- ListSets (displays available sets, which are not supported in this implementation).

These are best viewed in Firefox, which parses the XML tree nicely (don't use Chrome, which doesn't display the XML tree). There are also several links there to sites where you can learn more about the OAIPMH specification and function.

Other stuff

How streaming works in Podcast

The way Podcast conducts streaming is like this: When you click on a streaming link Podcast generates an .m3u playlist file on the fly, and hands it off to your browser. The playlist is just a text file containing a simple list of URLs for the relevant soundtracks (or other media) you requested.

Your browser will understand that it should launch your media player and start streaming and playing the soundtracks in the order that they occur in the playlist file. You are free to skip tracks and move through the playlist as you please.

The nice thing about this mechanism is that you can stream any kind of file with it. So long as your computer has an application capable of handling that file type, it will just work.

Technical considerations for streaming

Your server needs a fast internet connection to stream audio files effectively, especially if your audience is large. You should carefully consider bitrate when mixing down soundtracks or videos for publication. Obviously, audio recorded at lower bitrates is easier to stream.

As a general guide, 32 kbps is probably the best bitrate for voice recordings (16 kbps gives noticeably degraded but understandable quality, while it is hard to hear improvements beyond 64 kbps). For music 64 kbps is a minimum, 128 kbps is probably what you should aim for, depending on the bandwidth available to you.

If you want to stream video, well, that requires even more bandwidth! You will certainly need to optimise your videos for online viewing. Video compression is not in the scope of this manual, but there are plenty of tutorials available online. In a nutshell, you can compress video by reducing the frame size, reducing the frame rate, and by compressing both the audio and video tracks to various degrees. There are some handy programmes available that make compression quite easy, have a look at <http://www.handbrake.fr>.

You should also consider the bandwidth allocation from your web host. Podcasting is an insanely bandwidth-intensive activity if you have a large audience. Make sure you know what your bandwidth limits are and what charges apply if you exceed them!

Audio capture / editing software

Audacity is an excellent cross-platform and open source software for capturing, editing, mixing and compressing audio tracks. You can get it at: <http://audacity.sourceforge.net/>.

Podcasting clients

If you aren't familiar with podcasting, it's kind of like an internet radio programme that you can download on demand and play whenever you want. This is how it works: You install podcasting client software on your PC or smartphone. You subscribe to a regular podcast by entering the URL for the associated RSS feed into your podcast client, which will periodically check the feed to see if anything new has been published.

If a new soundtrack is available, your podcasting client will download it for you automatically. Good podcasting clients allow you to schedule updates so you always have the latest soundtracks available. It is particularly cool thing to set up on smartphones, as it means you always have something interesting to listen to when you are commuting or waiting for something.

For Android-based smartphones I recommend the BeyondPod client, which is excellent. For technology enthusiasts, I recommend checking out www.twit.tv for high-quality weekly podcasts (Security Now! and This Week in Google are well worth a listen).

Need help?

For general enquiries and support please post your questions in the ImpressCMS community forums at the link below. The author is a regular denizen there. Please do NOT email support questions to the author. Use the forums instead where everyone can contribute and benefit from solutions:

Want some custom modifications made to Podcast?

The author is happy to entertain requests for custom modifications to the Podcasting module. Also interested in developing additional OAIPMH-compliant modules for the ImpressCMS system. Contact simon@isengard.biz.

Annex 1: Programme submission form

Field	Recommended usage
Title	Name of the programme / album / collection.
Published by	Publisher of this album or podcast programme (the band, or company responsible).
Released	The year the album was published or the event was held. For ongoing podcasts, enter the start date or leave this blank (you can enter the publication date of each episode independently).
Description	A description or synopsis of the album / programme. Aim for one concise paragraph.
Compact view (yes/no)	Do you want to display this programme in compact view (a simple list of contents) or in expanded view (with descriptions for each soundtrack). If your soundtracks do not have descriptions, compact view is best. If they do, you can consider using expanded view.
Sort order for soundtracks (yes/no)	You can control the order in which soundtracks are sorted within the programme here. Podcasts with regular episodes are best sorted DESCENDING, ie. most recent soundtracks first). ASCENDING is best used on static collections of sound such as albums, or conferences where the talks follow a scheduled order. Note that the RSS feed and streaming links for this programme will also display soundtracks in this order.
Cover (upload)	Upload a programme logo or album cover here. There are restrictions on the maximum width/height/ file size of uploaded images. You can change these in the preferences section. Podcast will automatically rebuild thumbnail and screenshot images according to the dimensions you set in the preferences, which you can change at any time. Uploading a slightly larger image than you need gives you some extra flexibility if you decide you want to increase the size of displayed images later on.
Submission time	The date/time the programme was created.
OAI Identifier	Used to uniquely identify this record across federated sites, and prevents records being duplicated or imported multiple times. Do not change this value.

Annex 2: Soundtrack submission form

Field	Recommended usage
Title	Title of the soundtrack.
URL	A direct link to download the audio file, make sure you test it.
Author	Separate multiple authors with a pipe ' ' character. Use a convention for consistency, eg. John Smith Jane Doe
Contributor	Other contributors to the work, separate them with a pipe ' ' character.
Description	Description of the soundtrack. This is of most use in podcasts where it is common to provide a summary of the content, but for music albums you can leave it blank, in which case you should also set the programme (album) to use compact view. This will keep the layout looking good.
Format	The file format of the soundtrack. You can add more audio formats to this list by authorising Podcast to use them in System => Mimetypes.
File size	Enter in BYTES, it will be converted to human readable values such as KB and MB automatically.
Date	The publication date (official release) of the soundtrack.
Publisher	Name of the album, conference or event at which the recording was made.
Programme	The name of the podcast, album or collection the soundtrack belongs to. This is a select box of items available under the Programmes tab.
Language	Language of the recording, if any. Defaults to the default language setting in the module preferences.
Rights	The license under which this recording is published. In most countries, artistic works are copyright (even if you don't declare it) unless you specify another license. You can modify this list by editing items under the Rights tab.
Status	Toggle this soundtrack online (yes) or offline (no).
Federated	Syndicate this soundtrack's metadata with other sites via the Open Archives Initiative Protocol for Metadata Harvesting (enabled = yes, disabled = no).
Submission time	Date/time the record was submitted.
Submitter	Person who submitted the soundtrack.
OAI identifier	Used to uniquely identify this record across federated sites, and prevents records being duplicated or imported multiple times. Should never be changed under any circumstance. Complies with the OAI Identifier Format specification.

Annex 3: Open Archive submission form

Field	Recommended usage
Metadata prefix (read only)	Indicates the XML metadata schemes supported by this archive. Presently only Dublin Core is supported (oai_dc).
Namespace (read only)	Used to construct unique identifiers for records. Default is to use your domain name. Changing this is not recommended as it helps people identify your repository as the source of a record that has been shared with other archives.
Granularity (read only)	The granularity of record timestamps. The OAIPMH permits two levels of granularity, this implementation supports the most fine grained option (YYYY-MM-DDThh:mm:ssZ).
Deleted record support (read only)	This implementation does not support tracking of deleted records.
Earliest date stamp (read only)	The timestamp for the oldest record in your archive. This helps metadata harvesters frame appropriate OAIPMH requests (eg. they won't send requests asking for records that precede your earliest timestamp).
Admin email	The email address for the administrator of this archive. Be aware that this address is reported in response to incoming OAIPMH requests.
Protocol version (read only)	The OAIPMH protocol version implemented by this repository. Only version 2.0 (the current version) is supported.
Repository name	The name of your archive, by default your site name will be used, but you can customise this.
Base URL (read only)	The target URL to which incoming OAIPMH requests for your archive should be sent, which follows the pattern: <code>http://www.yoursite.com/modules/podcast/oaipmh_target.php</code>
Compression (read only)	Indicates what types of compression are supported by this archive in its responses to queries. Presently only gzip is supported.