

iTunes U Feedmanager

Software installation and usage documentation.

Overview	3
Terminology	3
About the iTunes U Feedmanager Software	4
<i>Developing a Web Application</i>	4
<i>CakePHP</i>	4
What iTunes U Feedmanager can do for you	5
Installation/Setup	6
<i>Requirements</i>	6
<i>Install the files</i>	6
<i>Setup the database</i>	6
<i>Setup CakePHP/Feedmanager Configuration Files</i>	7
<i>Setup the web server</i>	8
<i>Setup the scanning daemon</i>	9
<i>Setup the connection to XGrid</i>	9
Access and Usage	11
<i>XGrid jobs</i>	11
<i>Manage Items</i>	12
<i>Assign Items</i>	13
<i>Feeds</i>	13
Hosting of the Feeds and Assets	16
<i>Feeds</i>	16
<i>Assets</i>	16
Metadata mapping	17
<i>Importing from Podcast Producer Library</i>	17
<i>About the generated feed</i>	17

Overview

The iTunes U Feedmanager is a system based on web technologies that scans the library of an Apple Podcast Producer 2 Server and lets you comfortably create podcasting feeds that link to the multimedia files in that library. You can edit things like title and description of media items and feeds and also assign *Items* to one or more feeds dynamically through an easy to use web interface.

This software is explicitly meant to be used in conjunction with the Podcast Producer 2 Server. It is not possible to host files not stored in a Podcast Producer 2 library.

Terminology

A **Feed** in iTunes U Feedmanager is a collection of *Items* in a specific order. It also contains describing information about those *Items* like a title, description and category. It is also a XML document following the RSS2.0¹ specification expanded by the „itunes“² and „itunesu“³ namespaces which provide further options to describe objects in that feed. Inside the Feedmanager those properties can be edited and the XML files are dynamically created on demand from this data.

An **Item** is a representation of one or more *Assets* containing the same content (originating from the same source file) encoded in different file formats (for example video in HD and SD resolution plus an audio-only version). It also contains additional information of that *Item* like a title, description or a category.

An **Asset** is a representation of an actual media file.

¹ <http://cyber.law.harvard.edu/rss/rss.html>

² <http://www.apple.com/itunes/podcasts/specs.html#rss>

³ http://deimos.apple.com/rsrc/doc/UsingiTunesUPublicSiteManager/AboutTheFeedEditor/chapter_10_section_10.html

About the iTunes U Feedmanager Software

Developing a Web Application

The iTunes U Feedmanager is an application based on web technologies. It is programmed using the PHP language and runs on any php-enabled web server like Apache2 on Mac OS X Server systems. For data storage it uses a local Sqlite 3 database. You can access it using any standard web browser like Safari. The Feedmanager is developed as a web application because providing feeds is a web technology itself, so this saves an extra transition from an local app to the web. It also allows remote access from any workstation connected via network to the server, so you don't need to login locally on any machine or limit access to one user at a time.

CakePHP

The Feedmanager Software is based upon the *CakePHP* framework⁴ (Version 2.1.1). *CakePHP* is a programming framework built using the PHP programming language which provides a basic structure for web application developing. It already provides a lot of functionality to for example dispatch web requests or easily access database systems. It even has a command line interface which will come in handy later on.

Since providing feeds is a task closely related to web technologies, it's not a surprise *CakePHP* has a lot of needed technology already built in. The usage of prebuilt libraries speeds up the whole developing process a lot. And finally the architecture provided by the framework allows it to write clean and readable code which maintains expandability for eventual later needs.

⁴ <http://cakephp.org/>

What iTunes U Feedmanager can do for you

To manage content of an iTunes U site Apple provides the *iTunes U Public Site Manager*. This is a web interface similar to CMS (Content Management System), where you can manage the apparel of an iTunes U site in general and also manage what content appears where. But to actually provide media files to iTunes U, you have to create Feeds which contain some metadata (title, description and more) and links to the actual media files together with more metadata for those *Items*. iTunes U Feedmanager aims to provide those feeds and to provide a graphical interface to manage them.

With Feedmanager you can create feeds and enter their metadata in simple web forms. It automatically scans your Podcast Producer 2 library for *Items* and their *Assets*. The metadata of those *Items* can be edited and then dynamically assigned to one or more Feeds. The *Assets* represented by those *Items* then appear in automatically generated RSS 2.0 feeds separated after type (HD, SD or audio-only content). Feedmanager lists the addresses of those feeds for you to copy and enter in Apples *iTunes U Public Site Manager* where you create a *Collection* from such a Feed. This way you don't have to write or update such feeds by hand.

It tries to automate much of the workflow and provide an easy to use interface where manual editing is needed (in contrast to the CLI-tools you get with Podcast Producer 2).

Installation/Setup

For detailed information on the infrastructure provided by *CakePHP* and the necessary configuration (and also optional ways of installation/configuration), refer to the *CakePHP* documentation accessible at <http://book.cakephp.org/2.0/en/>.

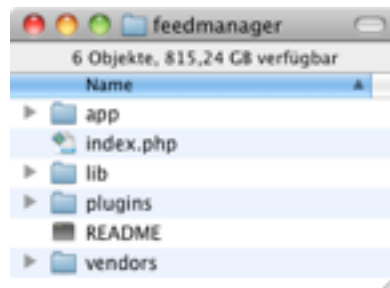
Requirements

The iTunes U Feedmanager is supposed to be running on the Mac OS X Server that also runs the Podcast Producer 2 Software (yet it is also possible to run it on another machine that also has access to the Podcast Library for example over the network, but such a setup isn't covered by these instructions). Since it is basically a web site, the Mac OS Web Service is required, but this should be already running when Podcast Producer was configured properly. It doesn't need lots of processing power or memory.

For access and usage any common browser can be used like Apple Safari or Mozilla Firefox. You can then either access the software from the same host or over any network.

Install the files

First you have to unpack the archive provided to any location on disk.



Inside the feedmanager folder you should find a folder called „app“, which contains the actual application code. The „lib“ folder contains the code provided by the *CakePHP* framework. Now you can move the whole „feedmanager“ folder anywhere appropriate, for example (on Mac OS X Server systems) in `/Library/WebServer/`. For security reasons it shouldn't be a location that is already accessible from the web. Yet the user the webserver runs as should have read-access to all the files and write-access to the following files and directories (and their subdirectories):

```
feedmanager/app/tmp/  
feedmanager/app/itunesufeedmanager.sqlite3db  
feedmanager/app/webroot/feedings/
```

Setup the database

Since the *CakePHP* framework works together with a variety of SQL database systems, you're free to choose between i.e. MySQL, PostgreSQL or SQLite. Included in the Feedmanager Software is a SQLite database file already setup with the right structure at `feedmanager/app/itunesufeedmanager.sqlite3db`. This might be the easiest way to start.

In case you want to use another database system, there is a schema file in `feedmanager/app/Config/Schema/schema.php`. You can use the Cake CLI (command line interface) to setup tables in your database according to this schema file. Refer to the *CakePHP* documentation under „Schema management and migrations“⁵ for that. In this case, you would also have to provide an alternative database configuration (see below) – for details, refer to the section „Configuration“⁶ in the *CakePHP* documentation.

Setup CakePHP/Feedmanager Configuration Files

To properly run *CakePHP* you have to adjust some configuration files provided inside *CakePHP*, precisely inside `/feedmanager/app/Config/`. To do so you have to have basic understanding of the PHP syntax.

core.php: This is the default *CakePHP* configuration file and shouldn't need any manipulation. Still it's never a bad idea to go through it and check to see if everything is okay.

database.php: Copy `database.php.default` to `database.php` and adjust the settings fitting your configuration. When using SQLite as recommended, you have to provide the full path to your database file here but can keep the rest as is.

feedmanager.php: Copy `feedmanager.php.default` to `feedmanager.php` and adjust the settings fitting your configuration:

- **Pcp.library-path:** Give the full path to your Podcast Producer Library here. This is the same as in the Podcast Producer settings in Server Admin and usually `/Library/PodcastProducer/Shared/`. Remember that the user the web service runs under has to have read and write access to this directory! There are instructions on how to do this in the configuration file.
- **urlprefix:** Give the prefix to the url this application should be accessed at. This is usually the hostname of the machine itself unless there's a need to rewrite the address i.e. when using a reverse proxy or similar.
- **asseturl:** This Url is used to create the Url the media files are accessible at. It provides the possibility to replace some special keywords as described in the configuration file. Usually this is the same as „urlprefix“ with the string `/assets/%asset_uid%.%asset_ext%` appended, but you can adjust the path for example if the files are actually hosted on another server which is directly accessible from the internet. Notice that in this case you have to take care of rewriting the url on access yourself!

The rest of the options are self-explanatory. They are mainly default values for strings that are „hard-coded“ into the generated feed. See the table at the end of this document for more information to these values.

⁵ <http://book.cakephp.org/2.0/en/console-and-shells/schema-management-and-migrations.html>

⁶ <http://book.cakephp.org/2.0/en/development/configuration.html>

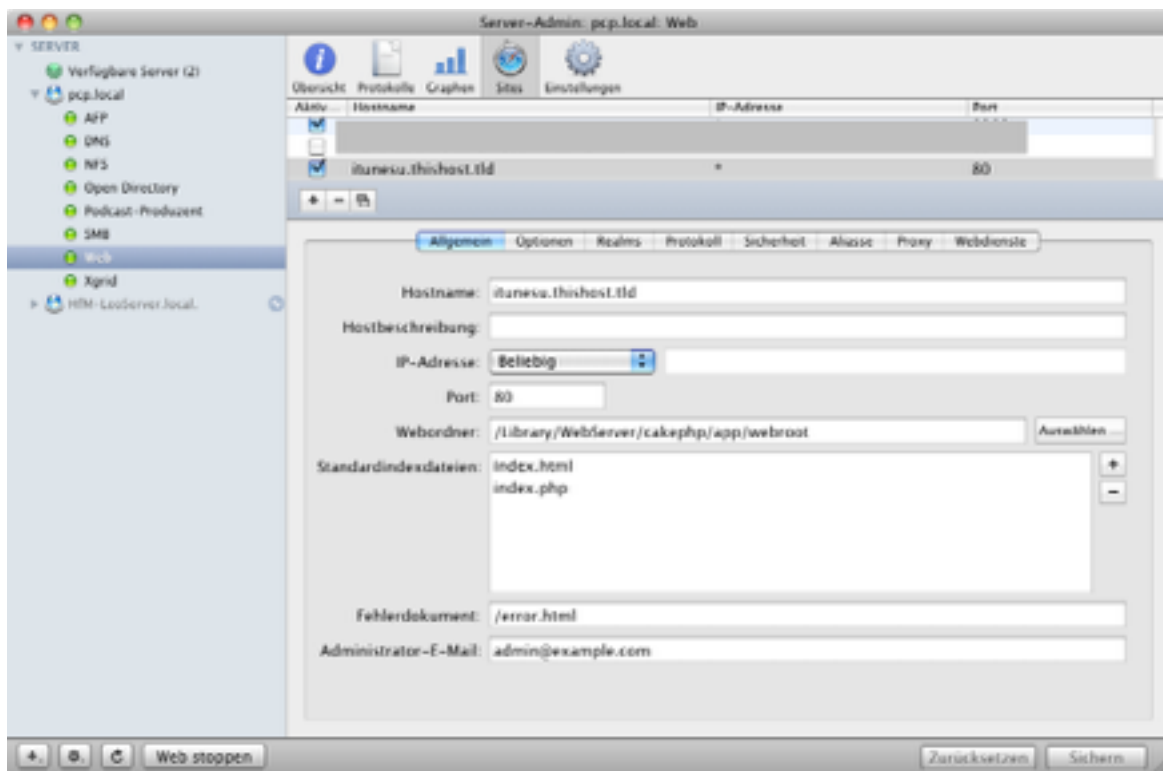
bootstrap.php: Copy bootstrap.php.default to bootstrap.php and adjust the settings fitting your configuration.

- itunesu-feed-languages: Give the languages you want to provide your feeds in here. Those become available as selectable options for each feed.
- itunesu-format-codes.hd: Give the format codes of files you want to provide as HD content here. Some default values are already there, but depending on the file formats you create with *Podcast Producer Server*, you might need to add entries here.
- itunesu-format-codes.sd: Same as above but for SD.
- itunesu-format-codes.audio: Same as above but for audio-only.
- itunesu-categories: Contains the categories and their codes available for use in iTunes U⁷. They are specified by Apple and shouldn't be changed.

Setup the web server

There are many ways to make *CakePHP* and the Feedmanager software work with a variety of web servers. For now I will describe the default installation using the Web service included in Mac OS 10.6 Server, which actually is an Apache 2 web server.

The easiest solution is to create a new Site in Apples *Server Admin* software under *Web* and choose the directory `feedmanager/app/webroot` as document root.



⁷ http://deimos.apple.com/rsrc/doc/UsingiTunesUPublicSiteManager/AboutCategoryPages/chapter_8_section_3.html#/apple_ref/doc/uid/iTUPSM-CH8-SW3

CakePHP takes care of the rest by including the right PHP files. There is also a `.htaccess` file inside the `webroot` directory which contains settings to beautify the URLs.

For other ways of installation see the *CakePHP* documentation under „Installation“⁸.

Setup the scanning daemon

The iTunes U Feedmanager has to regularly scan the podcast library folder for new content to include it into its own database. *CakePHP* provides a command line interface to do such tasks, called Commands. The script doing this task lies in `feedmanager/app/Console/Command/SyncpcplibShell.php`. This command can be called by opening a terminal session and executing the command `feedmanager/app/Console/cake syncpcplib` (from inside the directory containing the folder `feedmanager`).

This script scans the „Content“ folder inside the Podcast Producer library and compares it's content with Feedmanager's own database. *Items* or *Assets* that are not found in the database are being put into it. Also it checks the database for *Items* marked with the „deleted“ flag and deletes such *Items* from the file system and the database.

Since this script has to be run regularly and shouldn't be called manually each time, `launchctl` (an integrated part of Mac OS X) can be used similar to cron jobs. For that you have to install a special plist configuration file to `/Library/LaunchDaemons`. A default example file is provided in `feedmanager/app/Config/com.imwi.itunesufeedmanager.syncpcplibtocake.plist`. You have to copy this file to the `/Library/LaunchDaemons` and adjust the file paths according to your configuration. To start it the first time, you have to call

```
sudo launchctl load /Library/LaunchDaemons/com.imwi.itunesufeedmanager.syncpcplibtocake.plist
```

from command line once. But it should also be loaded automatically when the server boots.

As you can see in the plist file, the script writes to extra log files in `feedmanager/tmp/logs/`. These might help you debugging some problems.

Setup the connection to XGrid

The Feedmanager web application connects to the local *XGrid* server to retrieve information about running jobs and provide possibilities to pause, stop and delete running jobs. In any Podcast Producer 2 setup *XGrid* uses Open Directory/Kerberos for authentication, so the web application has to do so, too. For that to work you have to create a *keytab* file, which stores authentication information needed to access *XGrid*.

One way to do so is using the command line tool `ktutil`. Use the following commands:

```
ktutil
addent -password -p user@KDC.TLD -k 1 -e rc4-hmac
addent -password -p user@KDC.TLD -k 1 -e aes256-cts
wkt my.keytab
```

⁸ <http://book.cakephp.org/2.0/en/installation.html>

`ktutil` starts the tool itself and provides you with an extra interface.

`addent` adds an entry to an empty (not yet existent) file. As a user you can provide any user who has access to the *XGrid*, for example the default user created during automatic Podcast Producer Server setup. This user is usually called `podcastxgrid_some_uuid` and can be seen in the preferences pane of the Podcast Producer service in *Server Admin*. Also you have to replace `KDC.TLD` with your Kerberos domain controller, which is usually the same as your hostname. `addent` will ask you for the password of the given user.

`wkt` then writes the *keytab* file using the provided name (here `my.keytab`). Put that *keytab* file in the Config directory and make sure it belongs to the user the web server runs under and it has read-only access for only that user, allowing no one else to read it!

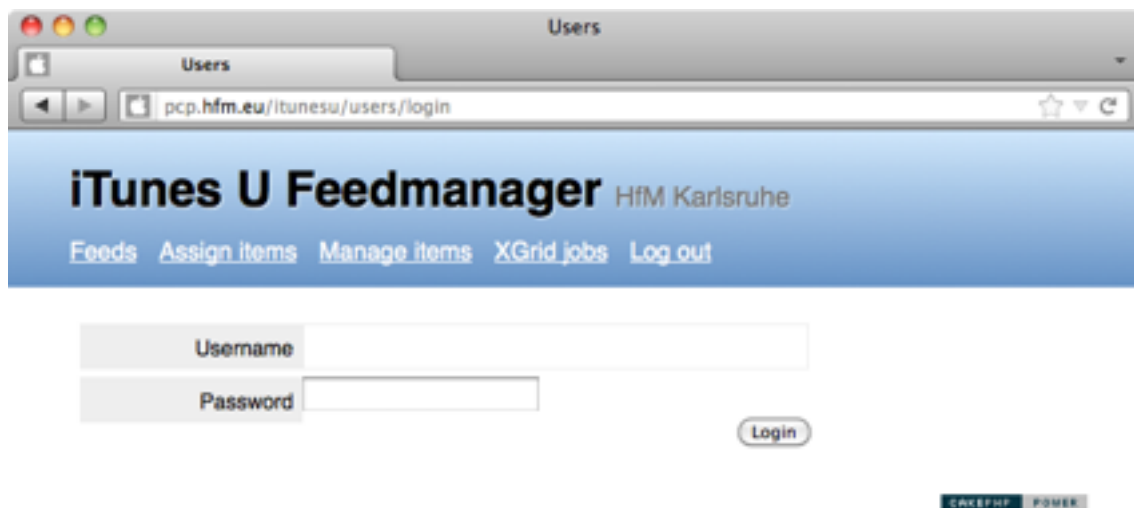
Access and Usage

Before you can access the Feedmanager software, you have to set a default password for the administrating user. In the database file provided there's already a user called „admin“, whose password you can initially set using the command line. When inside `feedmanager/app`, simply call

```
Console/cake users newpassword
```

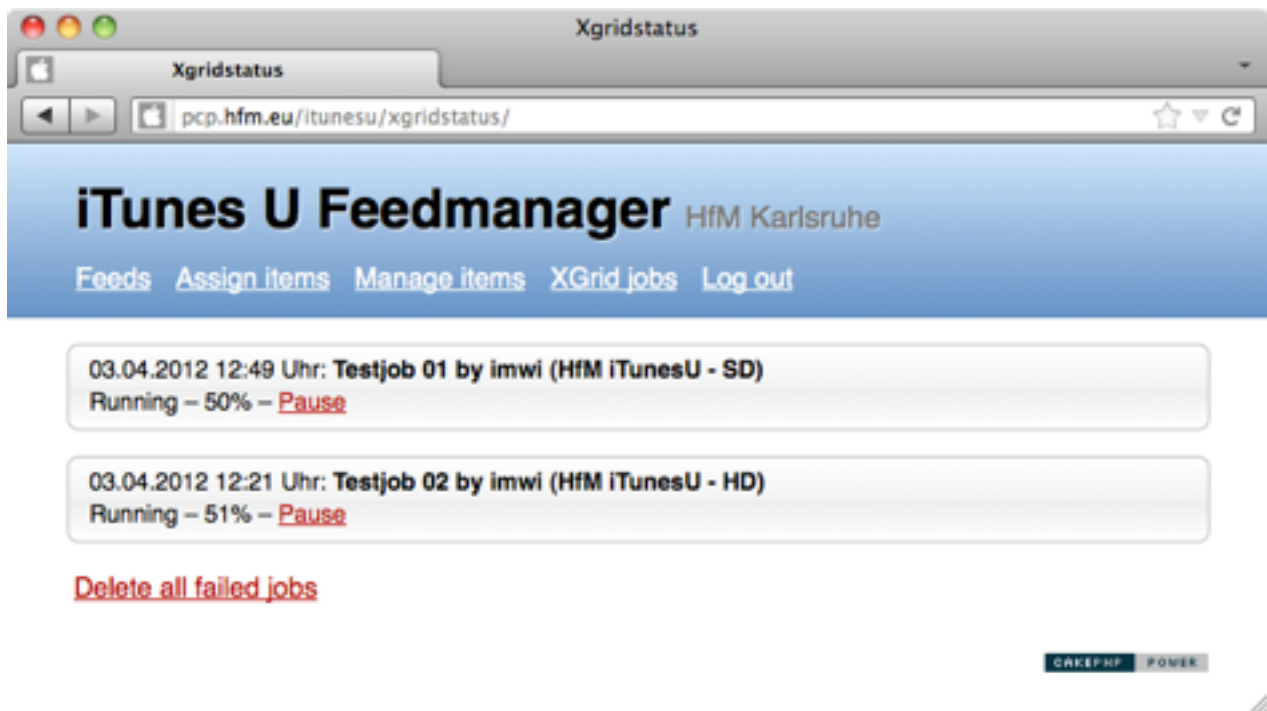
This will prompt you for the user whose password you want to change („admin“). Afterwards, give the new password twice. This updates the password and you will be able to login using the web interface.

The iTunes U Feedmanager is accessible using any standard web browser by pointing it to the configured address. It welcomes you with a login form where you have to provide your user and password you set up earlier.



XGrid jobs

On the XGrid jobs page, you can get an overview over running and failed *XGrid* jobs. You can see their status and Pause, Stop, Resume or Delete them from the list by clicking the links on each entry. You can also delete all failed jobs, which would stay on the list forever otherwise.



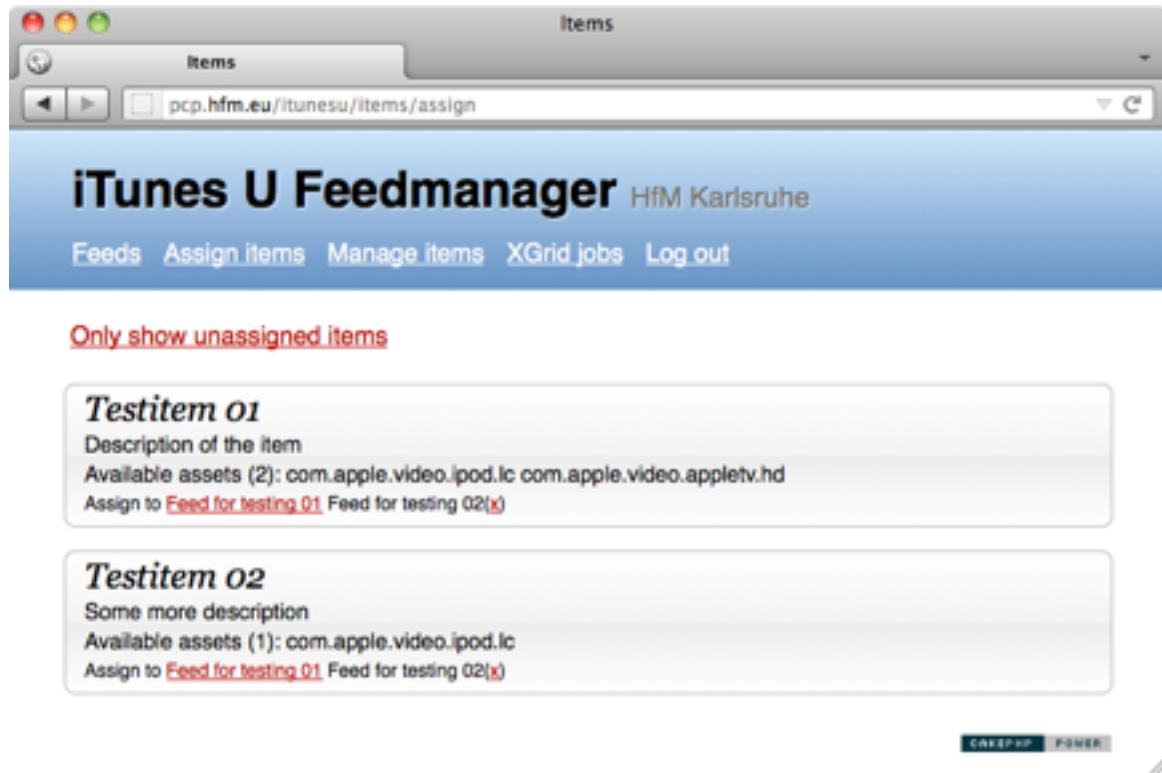
Manage Items

Here you can manage single *Items* that were imported into the Feedmanager. Newest *Items* are on top of the list. The list really contains all the *Items* and shows how many and what kind of *Assets* are assigned to that *Item*. Notice that *Items* might appear before their *Assets* because they usually need some time to render. Yet still you can already edit their title and description by clicking on „edit“. There's also the possibility to enter an author but that's optional.



Assign Items

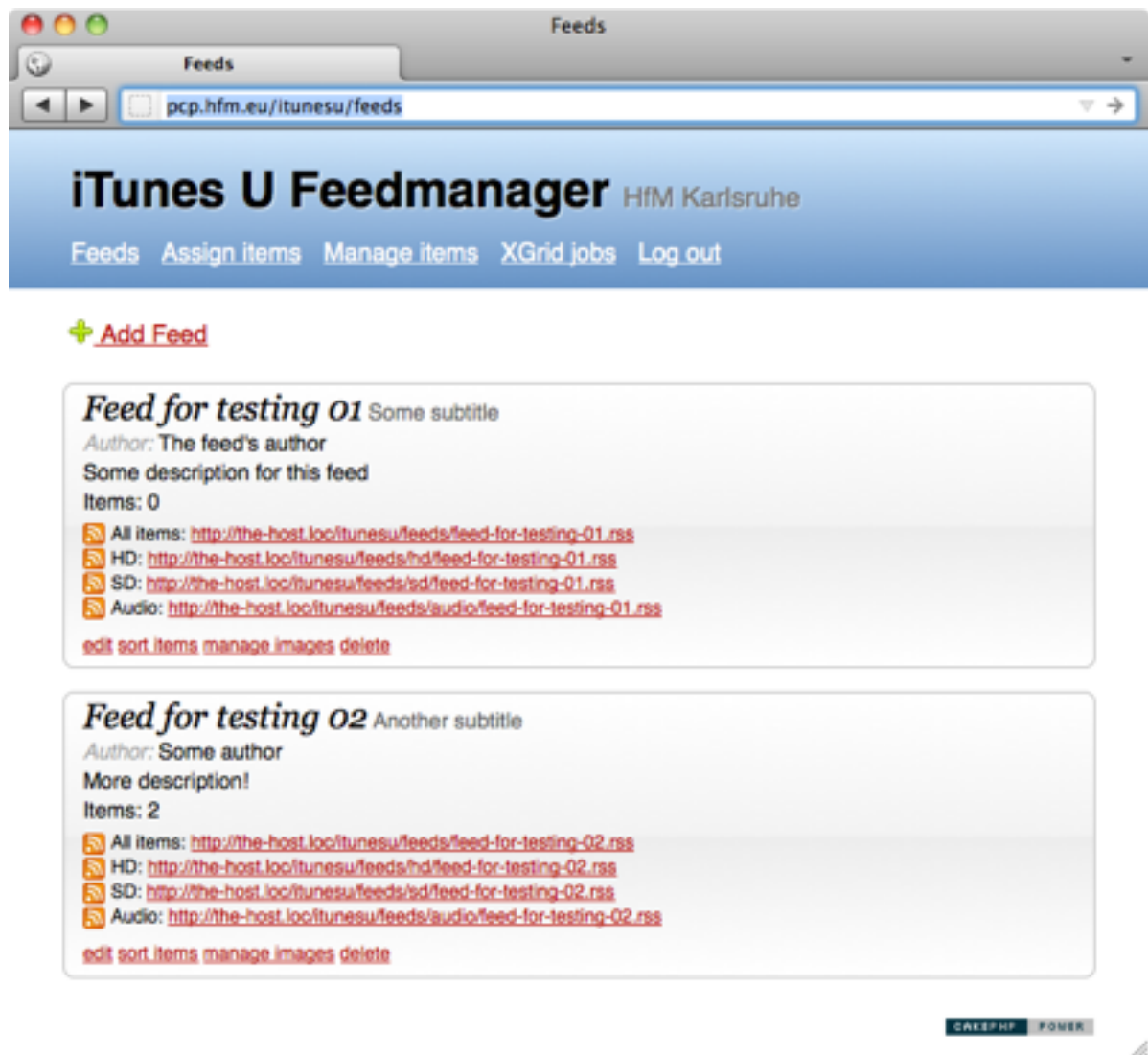
Under Assign Items you can assign each *Item* to one or more feeds. Once a feed has been created, it appears under each *Item* in form of a link. Clicking this link assigns an *Item* to that feed. You can remove that assignment by clicking on the little „x“ next to an assigned feed’s title.



Clicking on „Only show unassigned items“ gives a list with only *Items* that are not assigned to any feeds yet. That makes it easy to find *Items* that still need to be taken care of.

Feeds

Here you get a list of all the feeds created. You also get most metadata printed out (title, subtitle, author, description) and some additional information like how many *Items* have been assigned to it. You also get the addresses of the actual feeds you have to enter when creating a new *Collection* in *iTunes U Public Site Manager*. You can either choose a feed containing only a specific type of Media (HD, SD or Audio only) or use the „All Items“ feed, which could be useful when you want to mix contents of different kinds.



For existing Feeds there are several options:

Edit feed

This allows you to edit the feed's information. The fields are the same like when creating a feed, but there are two special options:

- *Don't update address*: Notice that the address of a feed which is usually generated from the title isn't being updated when changing the title unless you explicitly want to! In this case uncheck this checkbox.
- *Include audio-only assets in video-feeds*: Usually there are feeds for only HD, SD or audio files generated. But if you have video and audio *Items* in the same feed, you might want to include the audio *Assets* in your video feeds (HD and SD). In that case check this checkbox.

Sort Items

Here you can sort the *Items* manually. Usually *Items* appear in the feed in the same order as they are assigned to it, but in some cases this is not what you want. Then you can move the *Items* up or down by using the little green arrows next to an *Item*.

Manage images

Here you can upload or delete the image you want to assign to the feed. You can provide an extra image for each type of content (HD, SD, Audio) individually. If you don't do that, Feedmanager automatically falls back to the default image. The images are being stored on the Feedmanager server and linked to inside the created feeds.

Upload an image by selecting it from your hard drive and selecting the format you want to assign it to. Delete it by clicking the delete link below an image.

Delete the feed

Clicking on „delete“ deletes the feed and all of it's metadata from the database. Assigned *Items* are not deleted, though, only the assignments are being removed. This cannot be undone, so be careful!

Add a new feed

Last but not least you can also create a new feed by clicking on the „Add Feed“ link at the top of the page. This provides you with all the fields necessary. You can edit all of the informations you provide here later.

Hosting of the Feeds and Assets

Feeds

As already described, iTunes U Feedmanager provides the feeds needed in *iTunes U Public Site Manager*. You can obtain the address of such a feed in the Feeds page. Correctly configured these feeds must be accessible from the internet for Apple *iTunes U Site Manager* to read them. Generally you can open them with any common feedreader or podcasting tool like Apple iTunes. This way you can check if everything works properly. Validating a feed using automatic feed validators maybe won't work though, because not all iTunes- and iTunes U-specific XML tags are officially supported by RSS2.0.

The feeds are generated dynamically on access. That means that changes to those feeds (like adding or removing *Items* or altering its title) are being reflected in the corresponding feed instantly. But since Apple caches the feeds and only so often checks for updates, your iTunes U Site might take a while to reflect those changes.

Assets

Assets are by default hosted on the same server as the Feedmanager. But since the Podcast Producer Library usually isn't (and should not be) accessible from the web, Feedmanager has to deliver those Assets. It creates a special URL for each Asset and when accessed it gets the media file from from inside the Podcast Producer Library and delivers it to the client.

Notice that this technique may have caused (unconfirmed) problems working together with Apples content distribution network in some rare cases. It is not clear if this is a problem on Apples side or in the Feedmanager or somewhere in between, but intense testing is recommended when using this technique.

Alternatively, especially when hosting a big library of content, it could make sense to host media files on another host. Depending on your configuration, this might reduce network overhead and CPU time when delivering media files. In this case you let Feedmanager rewrite the urls to Assets so they point to the right address (in `Config/feedmanager.php`). But you will have to take care of eventual rewriting/redirecting of the urls when accessing them yourself.

Metadata mapping

Importing from Podcast Producer Library

When an *Item* gets imported from the *Podcast Producer Library*, Feedmanager tries to take over some data that is already there. Usually you provide this data when uploading a media file to a *Podcast Producer Workflow* for example using *Podcast Capture*. The **Title**, **Description** and **Publication Date** are being copied to the newly created *Item* in Feedmanager. Each *Item* also has an (optional) author, but this isn't being imported from the *Podcast Producer Library* because there it is only the short username of the user who uploaded the *Item* which often differs from the person who actually created the content. For the *Assets* connected to such an *Item* Feedmanager stores the format-code (i.e. „com.apple.video.apple.tv.hd“) and the mime content-type (i.e. „video/x-m4v“)

Title, description and publication date of an *Item* can be edited after it was added to the Feedmanager database.

About the generated feed

When generating a feed, Feedmanager uses metadata which was imported from the Podcast Producer Library (for *Items*) and entered manually (for *Feeds*) to fill required XML objects in the RSS Feed. It is important to know what data is presented where in iTunes/your iTunes U page. Interesting resources about feeds in iTunes/iTunes U are „Making a Podcast“⁹, „Using iTunes U Public Site Manager: Creating a feed“¹⁰ and „Using iTunes U Public Site Manager: Category Codes“¹¹. Below is a non-exhaustive list:

Feedmanager	Feed	iTunes U Page
Feed -> Title	<pre><channel> <title>content</title> <image> <title> content </title> </image> </channel></pre>	Collection Page: Title (first headline) Item list on front page: Column „Collection“ Downloaded Item: Album title It also appears as title (not name!) of the image assigned to the feed, but this has no effect in the iTunes U Site.
Feed -> Subtitle	<pre><channel> <itunes:subtitle> content </itunes:subtitle> </channel></pre>	The subtitle doesn't actually appear in iTunes U but is searchable by the global search function so it could contain tag-like words users might search for.

⁹ <http://www.apple.com/itunes/podcasts/specs.html>

¹⁰ http://deimos.apple.com/rsrc/doc/UsingiTunesUPublicSiteManager/AboutTheFeedEditor/chapter_10_section_10.html#//apple_ref/doc/uid/iTUPSM-CH10-SW7

¹¹ http://deimos.apple.com/rsrc/doc/UsingiTunesUPublicSiteManager/AboutCategoryPages/chapter_8_section_3.html#//apple_ref/doc/uid/iTUPSM-CH8-SW3

Feedmanager	Feed	iTunes U Page
Feed -> Author	<pre><channel> <itunes:author> content </itunes:author> </channel></pre>	Collection Page: Second headline Item list on front page: Column „Artist“ Downloaded Item: Composer Falls back to „feed.default-author“ preference (in Config/feedmanager.php) if none is provided.
Feed -> Language	<pre><channel> <language> content </language> </channel></pre>	Collection Page: Left column below feed image In the feed the language is coded as two letter country code so it appears internationalized in iTunes.
Feed -> Category of contained Items	<pre><channel> <item> <itunesu:category> content </itunesu:category> </item> </channel></pre>	Collection Page: Left column below feed image The category is a code, see „Category Codes“ - ally applied to all contained <i>Items</i> of that feed/ <i>Collection</i> .
Feed -> Description	<pre><channel> <description> content </description> <itunes:summary> content </itunes:summary> </channel></pre>	Collection Page: Description (above <i>Item</i> list)
Feed -> Keywords	<pre><channel> <item> <itunes:keywords> content </itunes:keywords> </item> </channel></pre>	The keywords also don't appear directly in iTunes U but are searchable. They are given to all contained <i>Items</i> of that feed/ <i>Collection</i> .
Item -> Title	<pre><channel> <item> <title> content </title> </item> </channel></pre>	Collection Page: List of <i>Items</i> , column „Name“ Item list on front page: Column „Name“ Downloaded Item: Name
Item -> Author	<pre><channel> <item> <itunes:author> content </itunes:author> </item> </channel></pre>	Downloaded Item: Artist Item list on front page: Column „Artist“ Falls back to feed author if none was provided.
Item -> Summary	<pre><channel> <item> <itunes:subtitle> content </itunes:subtitle> <itunes:summary> content </itunes:summary> </item> </channel></pre>	Collection Page: List of <i>Items</i> , column „Description“ (also pops up when hovered) Downloaded Item: Comment

Feedmanager	Feed	iTunes U Page
feedmanager.php -> „feed.default-author“	<pre><channel> <itunes:author> content </itunes:author> <item> <itunes:author> content </itunes:author> </item> </channel></pre>	See „Feed -> Author“ and „Item -> Author“ above. Appears only when no other author is provided.
feedmanager.php -> „feed.channel-link“	<pre><channel> <link> content </link> <image> <link> content </link> </image> </channel></pre>	Collection Page: Left column below feed image in the link box under „Website“ Should provide a link to a web site of your University/ Institution. The image link is not used in iTunes U.
feedmanager.php -> „feed.channel-owner.name“, „feed.channel-owner.email“	<pre><channel> <itunes:owner> <itunes:name> content </itunes:name> <itunes:email> content </itunes:email> </itunes:owner> </channel></pre>	This doesn't appear in iTunes U at all. It should provide name and email address of someone responsible for the feed.
feedmanager.php -> „feed.copyright“	<pre><channel> <copyright> content </copyright> </channel></pre>	Collection Page: Left column below feed image Short copyright statement for your content.
Publication Date (automatic)	<pre><channel> <item> <pubDate> content </pubDate> </item> </channel></pre>	Collection Page: List of <i>Items</i> , column released iTunes U orders <i>Items</i> after publication date when no order is provided otherwise. It Takes the date automatically from when it was sent to Podcast Producer Server. Also important for the „What's New“ box on front page.
Item order (automatic)	<pre><channel> <item> <itunes:order> content </itunes:order> </item> </channel></pre>	Collection Page: <i>Items</i> are ordered after this field. Order of <i>Items</i> can be edited by selecting „order items“ on Feeds page in Feedmanager.