

# HQmpd – Project Description

Anders Karlsson

*andekar@student.chalmers.se*

Tobias Olausson

*olaussot@student.chalmers.se*

September 7, 2009

The goal of this project is to show that Haskell can be easily used in practical applications by implementing an music player with a real-time decoder. The primary goal is to make this program execute fast, while hopefully keeping the elegant code style that Haskell is known for.

The style of the player will be client-server, with the server making sure that sound is outputted, keeping track of playlists and such, whilst the client may query the state of the server and/or sending commands to control output. For compability reasons, the player will implement the same protocol as mpd<sup>1</sup>. However, since mpd is lacking queue functionality, we will extend the protocol to include such a feature.

To accompany this player, and to minimize the need for external libraries, we will also implement a decoder for some music format(s), possibly mp3/ogg/flac or similar. This is especially interesting, since there is no real-time decoder for any of the formats written in Haskell to this date.

To make it possible for the user to both configure, and extend the functionality of the program, a configuration file, written in Haskell, is provided. If this file changes, the program will recompile itself the next time it starts. This behaviour is inspired from both XMonad<sup>2</sup> and Yi<sup>3</sup>.

If there is time at the end of the project, we will aim to write a small API for clients to use when sending commands. This API will most likely look exactly the mpd library available on hackage<sup>4</sup>, but with queue support. Together with this we would like to write a simple client like the command-line mpc utility.

---

<sup>1</sup>[http://mpd.wikia.com/wiki/Music\\_Player\\_Daemon\\_Wiki](http://mpd.wikia.com/wiki/Music_Player_Daemon_Wiki)

<sup>2</sup><http://www.xmonad.org>

<sup>3</sup><http://www.haskell.org/haskellwiki/Yi>

<sup>4</sup><http://hackage.haskell.org/package/libmpd>