Status report 20.02.06

Progress report

We have just started our third release cycle and we have implemented a simple audio engine able to play MP3 and XML parsing (SAX parsing) of the 2.02 standard. In our first release cycle we used GStreamer and REXML with Ruby, but were not happy with the result. In our second release cycle we reimplemented the functionality we made in the first release cycle, using C instead of Ruby as programming language. We are using mad, libao and libxml2 as our libraries, which seems to work nicely.

Our current work is to improve and expand our code, and add support for the new Daisy standard (ANSI/NISO Z39.86-2005).

Group cooperation

We have good communication and cooperation within the group, and we are pretty confident that the good collaboration will continue.

Report writing

We have not had so much focus on report writing up to now. Our focus in the early phase has been to explore good solutions and implement them. We feel that this method has given us the flexibility needed to quickly evaluate, implement, and possibly discard alternatives. In the time ahead, documenting our work will have a higher priority as we now to a larger degree have decided on specific solutions and need to create documentation for those solutions.

Problems

We have experienced a few problems this far.

- We had difficulty getting gstreamer to provide sufficient audio control. Also, in our experience, GStreamer is not well enough documented. The ruby gstreamer wrapper we worked against lacked both in documentation and basically did not give us enough control over the output to support our needs.
- The speed in our first XML parser caused us concern. While it was not critical, the Ruby implemented XML parsing was really slow.

To solve our problems, we decided to discard GStreamer and REXML for the Daisy engine and use well known C libraries like libmad, libao and libxml2. As we now would implement both audio and XML parsing functionality using C libraries, we decided not to use Ruby and implement everything in C. When we switched over from Ruby to C, the speed problem with the XML parsing solved itself.

Motivation

With this second release we feel a lot more confident in the project. It is a relief to discard REXML and GStreamer and our current implementation holds a lot more promise when it comes to maintainability, speed and control. In the future we hopefully will not experience the same problems with our choices as in our first release cycle. Now that these frustrations are behind us, our motivation is restored.

Øyvind Nerbråten	André Lindhjem
Kjetil Holien	Terje Risa