

# MusicalNodes

The visual music library

Lisa Dalhuijsen, Lieven van Velthoven

LIACS, Leiden University, Niels Bohrweg 1, Leiden, The Netherlands  
lisa@waks.nl, lievenv@gmail.com

supervised by Yanju Zhang, Fons Verbeek.  
yanju@liacs.nl, fvrbeek@liacs.nl

## Abstract

For this project we aimed at building a music management program for big, digital music libraries, out of dissatisfaction with well-known currently available software. We propose a system that solves some of the difficulties of organizing vast music libraries. The system introduces a new visual interaction style with the user's music collection. A physics system coupled to album's genre information allows the user to spatially order his or hers music on screen and make selections there. This, combined with the powerful search mechanism and the ability to add custom properties and information to music ensures rapid retrieval of music from even the largest of databases. User evaluations confirmed that all functions form welcome improvements over the standard text-based solutions currently available.

## Introduction

We are both heavy computer users and music lovers. We listen to music on our computers daily - if not all day - and have quite a big collection of mp3 files. We started thinking about this project out of frustration with existing software for playing music. While our collections are ever growing, it is increasingly hard to keep them well-organized. And even in an organized library it might be very difficult to find a specific song. We just miss a certain degree of flexibility and creativity in today's well-known music players. When software is being used on a daily basis, even minor inconveniences can lead to great dissatisfaction.

We wanted to propose and investigate a new way of how a digital music library looks and behaves.

## Problem definition

Firstly, in our opinion, existing music libraries like iTunes and Winamp are way too limited in their search and browse functionality. Both fully rely on a text-based approach, which means users with extensive libraries will have to wrestle through huge lists in order to find something. Of course the programs offer search functions, but for example the one in Winamp is useless unless you (partially) remember something from the name of the song, album or artist.

Imagine one wants to search for a reggae song

produced before 1980 by an artist whose name was probably "Bob", but is not too sure about that. Actually he doesn't know the title of the song either. In Winamp, for example, this is almost impossible. He could search for "Bob", but then may get hundreds of hits, while looking for only one. It can take quite some time to find that particular song that you want to listen to, because search functions offered by existing music players are (usually) not very flexible.

Our other big frustration was the way in which digital music is classified into genres. An mp3 file has so-called ID3 tags containing text, that can be edited by anyone. These tags contain information about the music, like the name of the artist and the year in which the music was produced. It can also contain the genre of the piece of music, like "rock" or "classical". However, our experience is that most music does not fit into one category. Most music is a mixture of different styles, for example funky jazz or electronic reggae. Therefore we wished it would be possible to categorize music more accurately, again allowing users to 'do it their way'.

The final main problem is that in available music programs it is hard to get a good overview of the music in the collection. Usually, one can browse by artist, album, etc., resulting again in huge lists of text, somewhere containing the music you are dying to hear. We feel that there is a lot of room for improvement in this area, since humans are just

not very fast at scanning large lists of text.

To ensure that we were not the only ones with this opinion, we sent out a questionnaire to people who listen to music on their computers regularly. We asked them how they organize their music and how they search for the music they want to hear. It turned out that almost everyone wished they had more flexibility in their music player, and that they would like to categorize their music in as many as possible ways. Some people told us they look for particular albums or songs in their collection, while other people look for a particular kind of music - for example relaxing, happy rock songs. This implies that the standard, pre-defined fields of ID3 tags are just not enough, since we found that different people want different ways of finding their music (The complete result of this questionnaire can be found in appendix 1.)

#### *User analysis*

Our target user group consists of music lovers who want better ways to search and organize their large collections. They use their computers intensively and listen to mp3 music daily, typically owning over 1.000 mp3 files, but collections of up to and over 50.000 files are no exception. Age is not so much a factor, but computer skills are. However, since our main goal was to build a solution that is easier to work with than the software people are currently using, any reasonably skilled computer user should be able to understand it.

#### **Approach**

So, our goal was to build a system that makes it easy to view, search and manage a music database. It is not (yet) intended as a music player, but a library in which users can easily select the music they want to hear and send it to their favorite player. In this section we will explain the different parts of the interface, and any novel aspects involved.

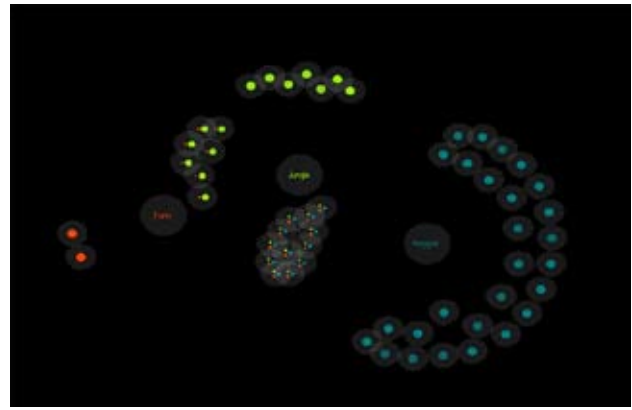
#### *Visualization*

Since humans are very good at recognizing visual properties and patterns, we thought a smart way of visualizing the collection would reveal much more information to the user in just a glance, than all familiar text representations. Below we explain how this was achieved.

#### *Nodes*

In our system, all music albums are represented by nodes: little 'balls' with different properties. These properties visualize various characteristics of the music. Currently, nodes are built up as follows:

- A transparent gray circle that lights up when the node is selected.



- One or more colored dots indicating the album's genres and age.
- When an album has multiple genres, colored dots show up for each genre. The size of the dots indicates the extent to which a piece of music belongs to each of the genres.
- The year in which the music was produced is represented by the dots' color saturation: the more saturated, the newer the music is.

Many other aspects of the music might be visualized in the future. For instance, the tempo of the music could be coupled to a 'jumping' movement of the nodes, shape could indicate 'mood', etc.

#### *Spatially organizing music*

To be able to easily organize their music, we were searching for a new intuitive way of doing this. Since people are very much used to organize things spatially (i.e. storing items in a cabinet, cleaning a desk), we wanted to use this for the music collection, starting with the genres and later extending it to other (scalable) properties. This way, people could ideally put their jazz and soul to left, electronic music to the right, and everything else between, for example.

#### *Genres*

This is easier said than done. We had a wish of not being limited a pre-defined set of genres, which was supported by many people in the initial questionnaires we sent.

Since albums can belong to any combination of genres with different ratios, the genre of an album is actually in multi-dimensional space. This position would then have to be mapped to a three-dimensional or even two-dimensional screen space, which is a well-known problem in mathematics. Many multi-dimensional scaling techniques exist in statistics and data mining, e.g. Principal Components Analysis. The problem with all these methods is that they almost never work perfectly, and are very inflexible to the user.

### *Physics*

So, to complete the visualization of the genres, an important innovation in MusicalNodes is the addition of a physics system. Musical genres are also represented by nodes, but attract the albums that belong to them. On the other hand, album nodes push each other away a little bit. This results in clouds of nodes on the screen, where a cloud corresponds to a collection of albums that belong to a specified genre. For example, you could have a jazz cloud, a blues cloud, a hip-hop cloud and so on. An album that belongs to multiple genres is attracted by multiple clouds. This way an album could be categorized as 30% near funk, 60% near jazz and 10% near blues. The Grandmaster Flash node would be at the center of the hip-hop cloud, while Nina Simone would be floating somewhere between jazz and blues.

### *Selection*

The genres can be dragged over the screen and placed wherever the user wants it, without effort. The albums immediately follow. When the user wants to hear music that is jazzy as well as bluesy, this is just a matter of seconds. He just drags the two apart from the rest, and simply select the nodes that float in between those nodes. Making selections of genres as easy as possible allows on the fly creation of playlists, according to the user's mood. In MusicalNodes, one can easily create a new genre, and assign it to a selection of music in the database. Assigning new or uncategorized music can be also done fast, as properties can be assigned to whole selections of music. In order to keep everything visible as the collection grows, the view can be panned in any direction and zooming is also available. In the future, different zooming levels might even show different levels of detail; i.e. genres will be visible from far away, albums from closer up and finally individual songs will show up.

### *Searching*

The second key part of MusicalNodes is its extensive search functionality and custom property editor. We wanted to provide great flexibility for the user to search through his database, so that the music can be searched or sorted by any property the user prefers; even properties 'invented' personally. Users should be able to easily modify anything in their library, and use anything to retrieve the music by. We learned by interviewing our target group that people indeed want to sort their music in different ways, besides obvious ways such as by artist or genre. One person thinks the mood of the music is relevant, while someone else finds the country of origin more important.

We incorporated this in the program by a means of firstly a search function that allows querying any property. Numerical properties also accept ranges

as queries. The search can be applied to the entire database, or to the currently selected music. This means multiple searches can be used sequentially to narrow down search results. A list tracks the queries (or filters) that are currently in use, and allow the user to delete or 'undo' a previous query.

The second aspect is the ability to create your own categories, besides obvious ways of categorizing such as by artist or genre.. MusicalNodes has a function that lets the user create a property, for example 'record label', and assign a value to it for any item in the database. Numerical properties also exist, allowing the user to select a value using a slider; for example 'happiness / sadness' of the music.

### **Results and evaluation**

MusicalNodes relies heavily on principles of the Gestalt theory, especially the law of prägnanz. This law says that people tend to order their visual experience in a manner that is regular, orderly, and simple. The mind groups similar elements into collections. This similarity might depend on relationships of form, color, size, or brightness. Spatial proximity of elements may induce the mind to perceive a collective or totality, and also elements with the same moving direction are perceived as a unit. If done right, the visual representation of the database can be interpreted faster and easier by our brains than large amounts of text. People don't need to 'translate' the text into information that is useful to them. We succeeded in using this principle in MusicalNodes to enable the user get an overview of a music collection at a glance, instead of having to read a list of items. In this system it is very easy to see certain groups of album as a whole, like albums that have the same genre or are from the same era in time.

Another main advantage of the visual representation is that it enables direct manipulation of the data. And while manipulating the genre-nodes' location, the mind instantly recognizes the albums belonging to the genre by color and movement.

### *User evaluations*

During the design process we tested our prototype on users twice. The first thing we wanted to test is the usability of the search function. We consider the search function a very important part of the program and we want to make sure that users can use it very quickly and easily. MusicalNodes contains use an iTunes-like search function that simply searches for a keyword in the entire database. Besides that, MusicalNodes offers an advanced search function that can be used to search for music, e.g. music from a specific period

or a specific artist. Most test persons understood how the filter function works quite naturally. They stated that they would really use the extended search functionality.

One problem we encountered during the evaluations was that the meaning of the terms used are not always clear to users, so we were happy that we could change that in our final prototype. All test users were very positive about the visual aspect of the program. They were convinced that it offers an advantage over conventional music players. Besides that, they enjoyed playing around with the nodes on screen.

### **Conclusion**

We believe that with MusicalNodes we have a great proof of concept for the ideas introduced by the program, which form a novel interaction paradigm. The key aspects of the program; the physics-based visualisation and direct manipulation of the music collection, the extensive search function and the custom properties all add up to form a powerful music library. Intended users who tested the system found it quite useful and welcome, and appreciated the novel aspects it features. The result is a piece of software that is easy to learn for beginners, and at the same time extremely flexible and powerful for more advanced users.

### **Discussion**

Ever since we first started working on MusicalNodes' design, we already felt it actually might work. But the prototype completely surpassed our expectations. The idea of using a physics system for the arrangement and clustering of music turned out to work better than we had hoped, as did all of the other key parts of the program.

We did realize this was going to be a pretty big project from early on in the first brainstorm sessions. Nevertheless, we really wanted to build this, if only because we 'needed' it ourselves. The implementation phase was a rocky road though. We encountered many, many difficulties while building the system, which were very time-consuming and reduced the amount of time left for fine-tuning and addition of extra features.

It might be good to realize that there are actually many different parts that were implemented to get MusicalNodes working, as it incorporates a full database, subsystems to handle 3D graphics, physics, advanced searching, a flexible property editor, etc.. For example, even a simple mouse-click on a node is not trivial; since the graphics are rendered in 3D (for speed considerations), the

coordinates of the mouse in 2D need to be mapped to 3D world coordinates first in order to check if the node was clicked.

Getting to learn the implementation specifics of C#, Direct3D, Windows Forms, ID3 tags, etc. took much of our effort during the implementation. We were (relatively) new to many of the libraries and techniques we use, and had to research a lot to be able to understand them. Also, sometimes strange errors would come up that were hard to resolve. But when finally all things came together, MusicalNodes worked exactly as we had imagined it would from the beginning, or even better.

### **6 Future Work**

We happily realize that that the MusicalNodes program could be much more extended than it is at this stage. We have lots of ideas that we still want to implement or try out, but could not within this period of time. Below is a selection of these ideas, of which some were already mentioned above.

- Assigning even more properties of the music to the appearance of the nodes.
- Adding a hierarchical 'level of detail' to the zoom function
- We also want to make it possible to arrange the nodes on screen differently. Instead of the genre-clouds, you could arrange the nodes by record label, happiness or country of origin.
- Linking subgenres to main genre-nodes, so that dragging one of those genres around will result in the related genres following.
- Integration with existing music player software
- And many, many other ideas about the user interface.

We have come to believe pretty strongly in this project, and we definitely plan to make MusicalNodes perfect and release it to the world. Our personal 'need' for good music management software will guarantee it...

## Appendix 1.

### User group questionnaire results

We interviewed 17 people by asking them the following questions:

How often do you listen to music on your computer?

- a) seldom
- b) quite often
- c) daily
- d) 24/7

What size is your collection?

How do you organize your music files?

- a) artists and albums are stored in different folders
- b) all files are in one big folder
- c) it's a mess

What program do you use for playing music?

What do you like about this program?

What don't you like about this program?

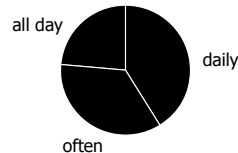
What kind of functions would you like to have in this program?

Music can be organized according to different criteria. In which of the following ways would you like to categorize your music?

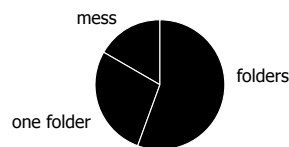
- a) genre
- b) tempo
- c) mood (happy, angry, sad and so on)
- d) acoustic / electronic
- e) year
- f) date added to program
- g) something else? ...

These are the results we got:

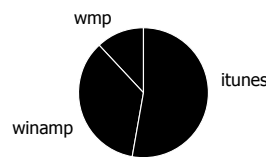
How often people listen to music on their computers



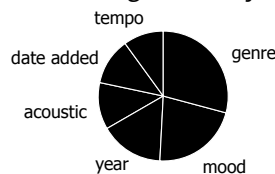
How these people organize their files



What program they use to play music



What categories they think are important



Suggestions we got on other categories

- label
- country of origin
- number of times played
- moment (time of day) of playing
- vocal / instrumental
- original / cover / riddim

The average number of mp3 files was about 6000, varying from 127 to 56,000 files.

Some people told us they look for particular albums in their collection, while other people prefer a random selection of music. In the latter case people want an 'intelligent' shuffle function, that plays songs with certain properties.