COSC460: Music Selection for Internet Radio

Alex Weiss
Department of Computer Science
University of Canterbury
New Zealand
Supervisor: Dr. Donald A. Adjeroh

November 3, 2000

Abstract

Music. Radio. The internet. Three very interesting fields which are linked in this project on internet radio stations. A comparative study of existing radio stations is performed. Weaknesses and problems in current internet radio stations, as exposed by the comparative study, are identified. We outline the general problems observed. In particular, focus is placed on automatic music selection. Our approach is based on the notions of music genre, popularity, repetition, catalogue coverage, style continuity and "music programme" generation. A simple internet radio station has already been implemented. The station serves as a platform to test our algorithms.

Contents

Chapter 1

Motivation and the problem

Some of us would have experienced the use of an internet radio, i.e. listening to a radio station on the internet using a computer. A computer is used to receive and play the audio stream, as opposed to the traditional radio context, where a receiver is used. Whereas in a traditional context, a radio station is usually physically located in one's geographical area and transmits the radio signals through the air as radio waves via a transmitter, in the case of internet radio, the station is located on some computer connected to the internet and the signal is sent to the receiver via the internet, all in digital form.

Although all that is strictly necessary to listen to an internet radio station is the audio client which lets one connect to the internet radio station's stream and listen to it, internet radio stations are always accompanied by a website. The website can contain extra items such as playlists, automated request mechanisms, programme schedules, song rating mechanisms, latest news and prize opportunities.

Internet radio has several advantages over traditional radio. It reaches a much wider audience, and allows many more possibilities than traditional radio. For example, a web-based interface could allow listeners to select their own music from the station's database, as opposed to traditional radio where listeners listen to what is played without any choice. Also, users could connect to the web to see what song is currently playing, what has been played and what is about to be played.

Internet radio can only become increasingly popular, as more people obtain get connected to the internet.

1.1 The problem of individual track selection

What must a radio station take into consideration when selecting what music it is going to play? The main thing is to play music that the station's listeners will appreciate. This will ensure the station keeps its existing listener base. This is the motivation behind one of the fundamental aspects of this project:

determining the popularity of tracks, artists and genres (items), and accurately reflecting changes in item popularity over time.

Of course, simply playing music a radio station's current listeners will appreciate is not the only consideration a station must take into account. A station may need to ensure that it covers a certain percentage of a catalogue (e.g., it may have a business agreement with SONY to cover one of SONY's music catalogues).

A station will also need to play new music or undiscovered older music which will add variety to their repertoire, will pleasantly surprise existing listeners and will hopefully attract new listeners to the station. This provides motivation for devising a method for selected these "surprise element" tracks.

Finally, as a variety of considerations (sources) must be taken into account when selecting a track, this provides motivation for developing a method that can handle multiple sources.

1.2 The problem of music programme generation

Rather than selecting the individual tracks each time we need to play some music, we will typically need to select different tracks to be played over a certain period of time, for instance a 60 minute programme. The problem of music programme (or sequence) generation then becomes how to arrange the selected individual tracks in a naturally flowing order. This is a combinatorial problem, with $n!$ arrangements possible if n tracks have been selected for the programme. In this project, we investigate how to use tempo transitions and transition tracks to address the problem.

Chapter 2

Background

2.1 Traditional vs internet radio

The following is a list of the main features of each method:

2.1.1 Internet radio

- Reaches a wider audience around the globe.
- More features are available (for example, through a web based interface to the internet radio station).
- Much cheaper to set up; no license for air waves is required.
- A listener is more likely to find a station matching their exact taste of music on the internet because there are many more internet radio stations accessible than local traditional radio stations.

2.1.2 Traditional radio

- A computer is not required (i.e. it can be used in almost any place such as cars or walkmans).
- Computer literacy is not required.
- More reliable (no dependence on the internet, local area networks, internet service providers). No network congestion problems.
- Quality of reception is guaranteed to be good providing the listener is within the geographical restrictions of the radio transmitter.

2.2 Analogies between traditional and internet radio stations

2.2.1 Tuners

Traditional radios have a tuning knob or some equivalent thereof. With the tuner, listeners can select the particular radio station they want to listen to. "Tuners" for internet radio stations are websites with lists of internet radio stations, usually sorted/grouped on some criteria such as genre or location. This allows users to browse through hundreds of stations. Several tuning websites exist.

With traditional stations, it is easier to locate all stations accessible to a user, as there are only certain frequency ranges within which a station can broadcast in order to be received by a home stereo. On the internet, one must rely on "tuners" as described above, or search engines to locate a station. However, there are many more internet radio stations accessible to a user than traditional stations, so while a user may find all the traditional stations he/she can access, this will be significantly fewer than the number of stations he/she will find and be able to access on the internet.

2.2.2 Requests

Requests to traditional radio stations are usually done via phone, fax or regular mail, though email is becoming more popular. Requests to traditional stations may often only be done during certain times of the day when the DJ's are accepting requests. All requests to traditional stations require some form of human intervention to process and initiate the request.

Internet radio stations can be set up to automatically process requests, requiring no human interaction.

2.2.3 Popularity

Traditional stations identify what their listeners find popular in limited ways. Listeners may phone in their requests during certain times of the day or place their votes for a top n countdown show. Traditional stations thus best gauge popularity from informal discussion with DJs/announcers and by analysing listener's votes for top n countdown shows.

Internet stations can obtain popularity information from their listeners in several ways. Users may vote for top n countdown shows or communicate with the DJs/announcers via email or live chat, as is the case with traditional stations. While informal discussion with a DJ/announcer is a fairly limited data gathering medium, internet radio stations also allow requests and ratings of currently playing songs to be made online, and have the user's request and rating information logged for later analysis. Internet radio stations can also keep a user profile of each user's likes and dislikes of particular tracks, artists and genres to help them determine popularity.

Both traditional and internet radio stations can use external sources such as top n charts to obtain popularity information.

2.2.4 Audience

We assume that stations playing similar genres/styles of music will attract similar listeners, irrespective of whether the station is traditional or internet based. For example, a station playing modern techno/dance music would attract mainly younger listeners while a station playing country/oldies would attract mainly older listeners. We also assume that while there are more internet radio stations than traditional ones, the mean listener–per–station figure would be higher in the traditional context.

2.2.5 User information

For internet radio stations, administrators can set up a logon page or an online questionnaire to gather demographic information about their listeners. Also, analysis of the internet addresses can reveal information about the location of its listeners.

In a traditional context, a website may also be used, but it is less likely to be as effective as in an internet radio station which is already internet based. Traditional stations can gain an informal feel for who their listeners are by which people ring up the station and talk to the DJs or announcers.

2.2.6 Access Restrictions/Control

Traditional stations have no way of restricting their users. An internet station can easily implement some kind of authentication scheme which means a user must first authenticate before being able to listen to a station. Although the purpose of such an access control mechanism may not be too obvious, it could have applications where there is need for restriction, e.g. in military applications (compare this with frequency-based access restrictions in traditional (not broadcast) radio communications).

2.3 Components of an internet radio station

An internet radio station is comprised of a server to stream the audio and clients (listeners) connecting to the audio stream. There may or may not be a separate "music server" streaming the music to the central server which then streams to all the clients. The architecture of the testbest internet radio station used in this project is shown in Figure 2.1. It runs an Icecast [1] streaming server on a Linux platform. Icecast streams *mp*3 [2] (MPEG Layer III) [1] audio streams.

[1]http://www.icecast.org
[2]http://www.mp3now.com

Readers may connect to the testbed internet radio station used in this project via the following addresses.

- audio stream — `http://132.181.8.3:8000`
- website — `http://132.181.8.2`

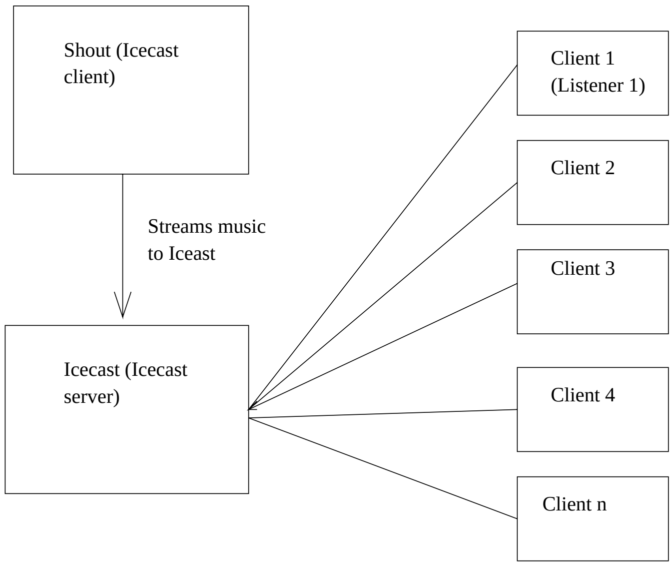


Figure 2.1: Icecast based internet radio architecture

2.4 Discussion

We have outlined the differences and similarities of traditional and internet radio stations. Readers are encouraged to visit an internet radio station (easily located via a search engine) if unfamiliar with them. Although traditional stations are mainstream at the moment, it will be interesting to see if and when internet radio overtakes traditional radio.

Chapter 3

Comparative Study

3.1 Introduction

A comparative study of 51 internet radio stations was performed. Internet radio stations were located via internet radio “tuners” — websites with lists of internet radio stations.

3.2 Survey

In order to gather more information, particularly on music selection, advertising and access control, a short ten–question questionnaire was emailed out to the stations in the comparative study.

The results of the study are summarized below. The raw data from the questionnaire can be found in Appendix A.

3.2.1 Streaming formats

There is more than one format which can be used to stream audio to listeners. We found the following four being used in our internet radio stations’ survey.

(Note some stations support multiple streaming formats).

format	ra	mp3	asx	qt4
number of stations	30	13	15	4

Key

ra	real audio (`http://www.realaudio.com`)
mp3	MPEG layer III (`http://www.mp3now.com`)
asx	windows media format (`http://www.microsoft.com`)
qt4	quicktime 4 (`http://www.apple.com/quicktime/`)

3.2.2 Multiple bandwidth support

Some stations offered the same stream at multiple bandwidths (i.e. low bandwidth and high bandwidths). This allows users on slower connections to still be able to listen to the internet radio station (albeit at a lower quality). Those with faster internet connections can connect to the higher bandwidth stream and receive the better quality audio stream. Of the 51 surveyed stations, 9 had multiple bandwidth support.

3.2.3 Playlist mechanism

By playlist, we refer to a display/list of what tracks have been played, what is currently playing and what is about to be played.

playlist	none	simple	medium	advanced
number of stations	39	2	9	1

Key

none	no playlist mechanism is in place
simple	shows what is currently playing
medium	shows what is currently playing, and what has been played
advanced	shows what is currently playing, what has been played and what will play next

3.2.4 Request mechanism

By request mechanism, we refer to the mechanism in place that allows users to request songs from the station's music database.

request mechanism	none	simple	medium	advanced
number of stations	41	8	2	0

Key

none	no request mechanism is in place
simple	user sends email requesting a track be played. There is no guarantee their request will be processed
medium	allows user to choose a track from the database
advanced	allows user to choose a track from the database, and has in place other mechanisms such as maximum requests per day or priority processing depending on user

3.2.5 Live/Pre-recorded

Pre-recorded means the audio being streamed is already stored on the storage device. Live means that the audio is being streamed as it is being generated.

This means that a DJ can be incorporated into a show or that live concerts may be streamed.

Of the 51 stations, 10 were pre-recorded while the remaining 41 were live.

3.2.6 Access Control

By access control, we mean the ability to "log into" the internet radio station via its website which would allow the user to do more than just listen to the internet radio station (which can be achieved without logging in via the station's website).

The vast majority of the stations had no access control mechanisms. Those that did (five stations) allowed users to do such things as receive news and information about the station, make requests, obtain additional information on the currently playing artist such as concert and album information, chat with the DJ and rate the currently playing song. No station had any form of hierarchy in their user structure to allow such things as priority processing of requests.

3.2.7 Advertising

Of the 51 stations surveyed, 50 were emailed the 10–question questionnaire (one station had no email address advertised). Of these 50, 17 responded.

Stations can advertise via banner advertisements on their website, advertisements in their audio stream and advertisements via email to their subscribed users, or they may not choose to advertise at all. The results from the questionnaire are as follows.

none	web	audio	email
9	5	5	3

3.3 Analysis/Discussion

From the comparative study, we observed that most stations had no request mechanisms. This means they have no way of assessing what songs their listeners liked and disliked apart from general email feedback. A few stations did have mechanisms to rate the currently playing song (usually on a 5 point Likert scale [4]). A Likert scale ranges from strongly disagree to strongly agree. With the absence of such an agreement, this leads us to wonder how stations determine the popularity of their music, and hence what to play. In the next chapter, we examine the problems of popularity and music selection in depth.

In this project, our main focus is on the aspect of music selection and sequence generation. This will be affected by the request mechanisms available and other sources of information used by the radio station. The generated sequence can also be used in constructing the playlist.

We found access control mechanisms to be lacking in most stations. Those stations that did have some form of access control had simple implementations.

This means users are unable to maintain a profile of artists/tracks/genres of music they like and dislike (which would help the station in selecting its music). It also means there is no way to give priority processing of requests to more important listeners (e.g., a university campus based internet radio station may give higher priority to postgraduate students, medium priority to undergraduates and low priority to external users).

Chapter 4

Music Selection

4.1 Background

Radio stations (both traditional and internet) require some form of music selection algorithms. However, the algorithms are typically proprietary, thus hidden from the public eye and academic circles.

Relevant works in the area of music selection include [9], [12] and [5]. In [9], a combinatorial approach to music selection is outlined. Popularity is not a major focus in this paper. [12] presents a technique for making personalized recommendations from any type of database to a user based on similarities between the interest profile of that user and those of other users. It presents a method for gathering popularity information about artists via a user profile mechanism. We adopt this user profile mechanism and incorporate it into our popularity figure system. We use the same base mechanism, but also add support for tracks and genres. In [5] music selection was performed by considering the mood of the user: cheerful, romantic, calm, sad and curious, and the location of the user (e.g. home or office), but is targeted at single user systems, not a group of listeners.

Our approach combines popularity, catalogue coverage, style continuity and multi-user dimensions into the music selection process.

4.2 Music selection vs music retrieval

It is important to note the distinction between music selection (which we focus on in this project) and music retrieval.

Music retrieval attempts to retrieve an audio sample based on some search query. The search is based on information extracted from *inside* the audio sample. For example, one method of retrieval described in [6], attempts to match tunes from its database with the acoustic input of the user; the user sings a few notes into his/ her microphone and the search engine attempts to match the notes sung by the user with songs in its database. Another method, proposed in [2], matches the query based on the inputted rhythm. Methods exist to extract speech from audio samples, and then provide speech to text transcription, allowing users to query based on words in the song/audio sample. For an overview of audio retrieval systems refer to [3]. A system for automatic genre extraction from a music track is described in [14], while [15] presents another general work on audio classification and retrieval.

Music selection, on the other hand, does *not* look inside the audio sample, and a track is not selected due to a specific query by a user. In this project, a track is selected based on its popularity, catalogue coverage, surprise element or other factors (these selection criteria are further expanded upon in later sections).

4.2.1 Other applications

The problem of music selection is not only relevant to internet radio stations. Other areas where automated music selection could be used include:

- traditional radio stations
- background music in shops, shopping malls, telephone holding systems
- clubs (automated DJs)

4.3 The music selection problem

We address the problem of music selection in an internet radio framework. In everyday radio broadcast, music that is played is based on popularity. If people enjoy listening to a track it will be played. In contrast, if nobody enjoys listening to a particular music, it will rarely or never be played. The problem then is how to know what people would (or would not) like to listen to. That is, how to determine the popularity of a particular title (or track), or of a certain artist.

In addition to popularity, other factors such as tempo, era and genre will be considered in the music selection process. For the purpose of this project, we will be manually classifying each track according to its tempo and genre, although work by other parties is in progress to automatically extract tempo [11] and genre [15][14] from a given music track.

4.4 The general approach

In selecting the appropriate music tracks, many criteria must be considered. These include track/artist/genre popularity, catalogue coverage, playing a variety of new, yet hopefully appealing music that listeners have not heard before, supporting local bands, not repeating the same tracks too often, etc. Each criterion (or factor) will have its own weighting as determined by the station administrators. The higher the weighting on a factor, the more likely a track is to be selected due to that factor. For example, if a high weighting is placed on popularity, a track is more likely to be selected due to its popularity. Conversely, if a high weighting is put on catalogue coverage, a track is more likely to be selected because it is in the catalogue, even though it may not be popular.

Musical genre trees are used to organize and structure the music collection owned by a radio station into genre and to help determine popularity.

A music programme is a well organized sequence of music tracks/titles, and hence, is different from a random sequence of individual tracks. However the generation of such organized sequences will rely on some method to select the individual tracks that will make up the sequence. Although individual selection is a problem on its own, our focus is on music programme generation, not individual track selection. A programme could consist of a top ten countdown of the station's top ten tracks, an hour's worth of the station's most popular rock, jazz and pop songs, or an hour's worth of little known but hopefully soon to be appreciated music. It is important to use programmes as they are more natural than a random sequence of individually selected tracks.

As we are using programmes, it is important to make the programme balanced and to ensure smooth transitions between tracks.

The following sections expand upon the general approach.

4.5 Musical genre trees

Musical genre trees allow us to organize and structure all the music titles owned by a radio station by genre. The tree could range from a fairly general tree encompassing varying genres for a "play a bit of everything" – type station, to a very specific tree for stations playing only one genre, or a subset of one genre. Figure 4.1 is an example of a general genre tree, while Figure 4.2 shows an expanded tree for a particular genre (electronica).

Musical genre trees also help to determine the popularity rating (further developed in the next section) of each genre, by propagating values from the leaf nodes up through the tree.

Trees are used instead of graphs because they make it easier to propagate popularity figures.

Tracks/artists may be classified under two or more genres if their style of music falls under a wider umbrella than one genre. Thus, instead of having cycles in our representation, we might have the same title/artist appearing in more than one leaf node.

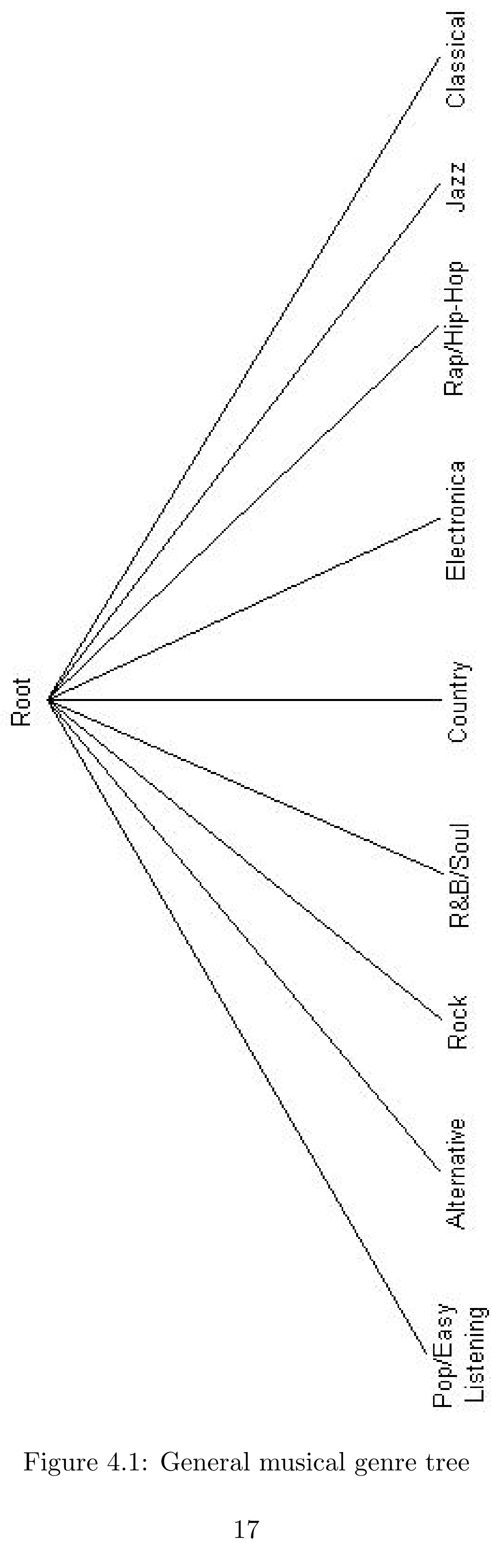


Figure 4.1: General musical genre tree

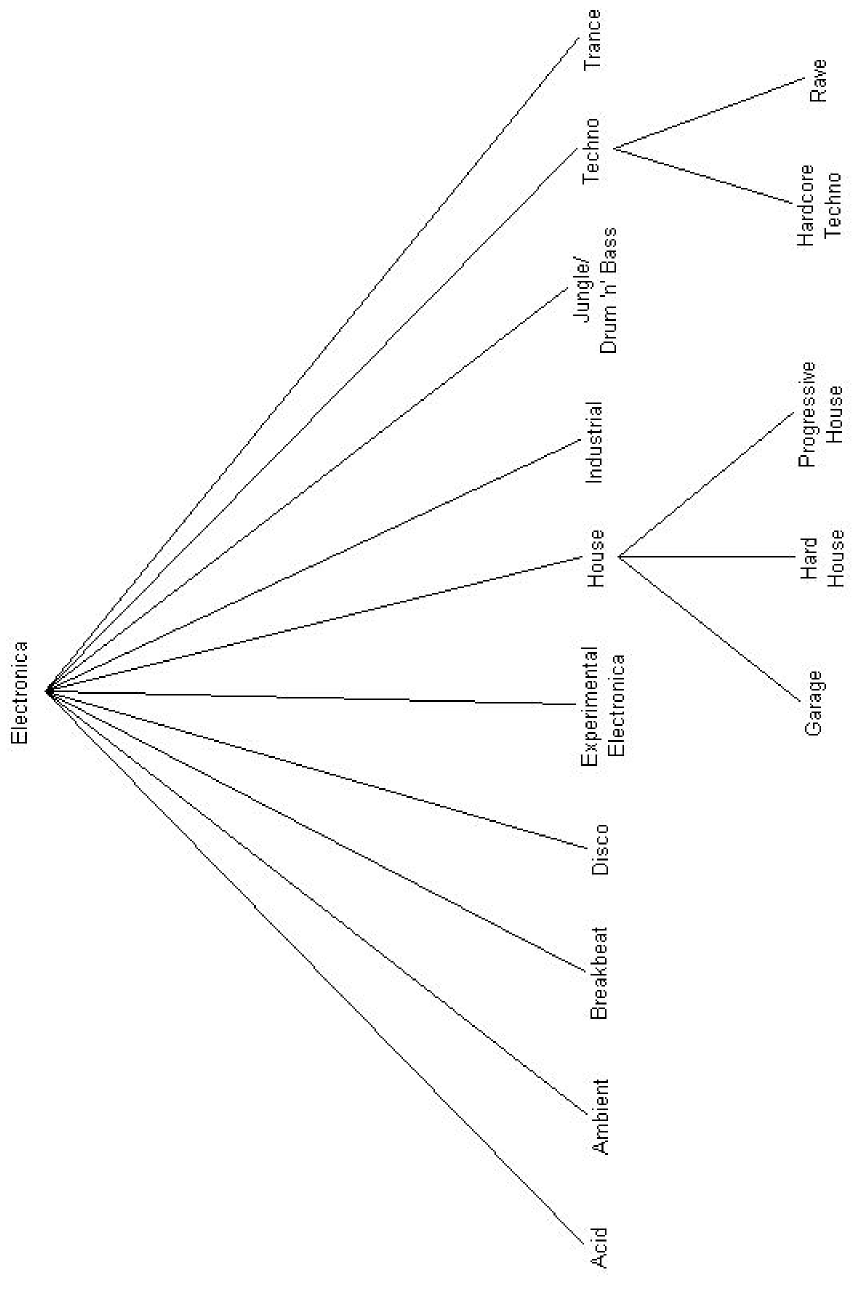


Figure 4.2: Specific musical genre tree (source: `http://www.mp3.com.au`)

4.6 Popularity

Popularity plays a key part in the music selection process. The popularity of tracks, artists and genres is gauged on a scale ranging from 0 (lowest) to 1 (highest).

4.6.1 Sources of popularity

User profiles

Users (i.e. listeners) of the station will each maintain a profile of the music they like (and dislike). This will be achieved by rating on a seven point Likert scale a selection of tracks, artists and genres. We use a seven point scale because studies show that the reliability of data does not increase substantially if more than seven choices can be made [10]. The artists rated will be comprised of a "core" selection, with the remaining being randomly selected. By core, we refer to a selection of well known artists that most people will know and whose music is likely to be played on the station due to their empirically known popularity. This will ensure commonality between different user profiles. Users will be periodically asked to update/review their profiles.

Popularity figures generated from user profiles will be denoted pf_1.

User requests

Users will be able to select tracks they wish to hear from the station's music database. Selecting a track will increase the track and associated artist and genre(s) popularity.

Popularity figures generated from user requests will be denoted pf_2.

User online ratings

Users are able to rate the currently playing song online, using a five point Likert scale (to increase or decrease its popularity rating).

Popularity figures generated from user online ratings will be denoted pf_3.

External sources

The sources so far described can be said to be internal to the station. They are based on users who are currently listening to the station.

We need external sources to incorporate other's views in order to moderate the ratings and to reflect the general trends of track popularity.

External sources include such things as third party top n charts, and what other radio stations are playing.

Popularity figures generated from external sources will be denoted $pf_{4,i}$, where i denotes the ith external source.

4.6.2 Combining popularity sources

Each source is given a weight, with the sums of all the weights totalling one. The weightings of each source are left to the discretion of the station administrators, however, we recommend giving most of the weighting to user profiles, user requests and the rating of currently playing song. This is because one should be primarily interested in ratings influenced by one's own listeners rather than by external sources. However, the external sources can also be used to attract new listeners.

For the four popularity sources described above (user profiles, user requests, user ratings and external sources), we assign weights w_1, w_2, w_3 and w_4 respectively.

The popularity of an item, denoted by $pop(\lambda)$, where λ refers to a track, artist or genre, is given by:

$$pop(\lambda) = w_1 pf_1 + w_2 pf_2 + w_3 pf_3 + w_4 pf_4$$

where pf_1 to pf_4 are the popularity figures of the four sources, as described in the previous subsections.

4.6.3 Calculating the popularity figures

User profiles

We use a 7 point scale. This translates to:

Likert scale	Raw popularity
1	0
2	0.1667
3	0.3333
4	0.5
5	0.6667
6	0.8333
7	1

A record of ratings for each artist, track and genre is kept. Note that if an item (track, artist or genre) is not given a rating, the non–rated items are not used in the calculations.

The pf_1 of an item is calculated as follows:

$$pf_1(\lambda) = \frac{\sum_{i=1}^{n} (R_i)(\gamma_i)}{n}$$

where R_i refers to the ith user profile rating of a particular item, and γ_i refers to that rating's associated raw popularity.

If we consider the specific example of the artist "Billy Joel", let us assume six people have rated him with the following: 7, 6, 6, 5, 3 and 2. This gives us the following mean popularity figure for user profiles:

$$\begin{aligned} pf_1 &= \frac{1(1) + 2(0.8333) + 1(0.6667) + 1(0.3333) + 1(0.1667)}{6} \\ &= 0.6389 \end{aligned}$$

If a seventh user rates Billy Joel a 6, the new pf_1 would be:

$$pf_1 = \frac{0.6389(6) + 0.8333}{7} = 0.6667$$

The same method would be used to calculate artist and genre popularity.

User requests

A record of each track's requests are kept. The more requests a track has, the more that track's popularity and that track's artist popularity will be positively influenced by this popularity source. The pf_2 of an item is determined by:

$$pf_2 = \frac{x}{n}$$

where n is the total number of requests made and x is the total number of requests made for that particular item.

A track's request record is reset to 0 if it has not received a request for a specified time period as determined by the radio station administrators (e.g. two weeks). n is altered accordingly. This ensures that tracks which are no longer popular do not get an unrepresentative popularity gain from this source based on requests made previously.

User online ratings

We use a five point Likert scale ranging from 1 to 5. Bearing in mind that 0 corresponds to no popularity while 1 corresponds to maximum popularity, the five points on the Likert scale translate to:

Likert scale	Raw popularity
1	0
2	0.25
3	0.5
4	0.75
5	1

We choose the 0, 0.25, 0.5, 0.75, 1 scale as opposed to the 0.2, 0.4, 0.6, 0.8, 1 scale a positive bias is not introduced into our system. By positive bias, we mean that there are more options which map to a raw popularity rating above 0.5 than below. A negative bias means the opposite. With users having a tendency to rate only music they appreciate, a positive bias in the rating would accentuate this inequality even further.

In plain English, the 5 points on the Likert scale could correspond to the following question and answers.

Question: Please rate this song according to the following scale:

Likert scale	Options (answers)
5	I love it
4	It's really good
3	It's OK, I don't mind listening to this song, but on the other hand, I don't mind not listening to it either
2	I don't really like it
1	I hate it

A record of the user rating of each track is kept. The pf_3 of an item is calculated as follows:

$$pf_3(\lambda) = \frac{\sum_{i=1}^{n} (R_i)(\gamma_i)}{n}$$

where R_i refers to the ith entry in the record of user ratings for a particular item, and γ_i refers to that record entry's associated raw popularity.

For example, let us assume seven people have rated "Billy Joel – Pianoman" 5, 5, 5, 4, 4, 4, 3.

This would correspond to a mean popularity figure for user ratings of:

$$pf_3 = \frac{3(1) + 3(0.75) + 1(0.5)}{7} = 0.8214$$

We now know the popularity due to source 3 is 0.8214. Let us now suppose the weights and corresponding popularity of each source, denoted (w_i, pop_i) are (0.5, 0.7), (0.2, 0.8), (0.2, 0.8214) and (0.1, 0.5572) respectively. This would mean that in our example, the popularity figure of "Billy Joel – Pianoman" would be 0.73, as shown:

$$\begin{aligned} pop(\text{Billy Joel – Pianoman}) &= w_1 pf_1 + w_2 pf_2 + w_3 pf_3 + w_4 pf_4 \\ &= 0.5(0.7) + 0.2(0.8) + 0.2(0.8214) + 0.1(0.5572) \\ &= 0.73 \end{aligned}$$

Now, let us assume a new online user rates "Billy Joel – Pianoman" a 3 on the Likert scale.

We recalculate pf_3:

$$pf_3 = \frac{0.8214(7) + 0.5}{8} = 0.7812$$

We now recalculate the raw probability of "Billy Joel – Pianoman":

$$\begin{aligned} pop(\text{Billy Joel – Pianoman}) &= 0.5(0.7) + 0.2(0.8) + 0.2(0.7812) + 0.1(0.5572) \\ &= 0.7220 \end{aligned}$$

As we see, the popularity of "Billy Joel – Pianoman" has dropped from 0.73 to 0.7220 in our example after the user rated it 3 on the five point user rating scale.

Although our example demonstrated the calculation of the popularity figure for a track (Pianoman, by "Billy Joel"), the same method would be used to calculate the popularity figure of the track's associated artist ("Billy Joel", in this case), and the track's associated genre(s).

External sources

As with user ratings, we want to avoid positive biases in our system. We therefore use the following formula to determine an item's popularity figure from a top n chart:

$$pf_{4,i} = \frac{n - x}{n - 1}, for\ n > 1$$

where n is the number of items in the chart, and x is the chart ranking of the item in question.

We divide by $n - 1$ as opposed to n so that that there is always the same number of ratings above and below 0.5.

For a top ten chart, this formula translates to the following raw popularity figures:

Chart rank	Raw popularity
1	1
2	0.8889
3	0.7778
4	0.6667
5	0.5555
6	0.4444
7	0.3333
8	0.2222
9	0.1111
10	0

As 10 is an even number, there is no middle value, and thus, no rank corresponds to a popularity figure of 0.5.

The $pf_{4,i}$ of an item is calculated as follows:

$$pf_{4,i}(\lambda) = \sum_{j=1}^{n}(w_j)(\gamma_j)$$

where w_j is the weight associated with the jth component of external source i, and γ_j is the raw popularity of the jth component of the external source i.

In the case of top ten charts, let us consider an example of three top ten charts. Suppose the track "Robbie Williams – Strong" is ranked 6, 7 and 4 on three separate charts. The popularity figure of this track based on these three charts would be given by:

$$\begin{aligned} pf_{4,1} &= \frac{\frac{10-6}{9} + \frac{10-7}{9} + \frac{10-4}{9}}{3} \\ &= 0.4815 \end{aligned}$$

Note the above example assumes equal weights for each top ten chart. It may be that the first chart is more respected in the music industry than the other two in which case station administrators may wish to give it a bigger weighting, eg $w_{chart1} = 0.6, w_{chart2} = 0.2, w_{chart3} = 0.2$.

The revised popularity figure $pf_{4,1}$ for "Robbie Williams – Strong" is:

$$\begin{aligned} pf_{4,1} &= 0.6\left(\frac{10-6}{9}\right) + 0.2\left(\frac{10-7}{9}\right) + 0.2\left(\frac{10-4}{9}\right) \\ &= 0.2667 + 0.0667 + 0.1333 \\ &= 0.4667 \end{aligned}$$

We note the new $pf_{4,1}$ reflects chart one's ranking more strongly now.

In considering what other radio stations are playing, this may simply mean increasing the popularity of the given artist(s) or track(s) that the other stations are playing.

External top n charts will only be relevant for contemporary music. This means that non-contemporary music should not be included when using top n charts to influence popularity figures. It also means that we need to consider other methods that will be able to handle non-contemporary titles/artists. One such method would be to give 0 weighting to external sources using contemporary top n charts, but increase the weightings of other sources.

4.6.4 Updating popularity figures

As time passes, the popularity of a track will naturally decrease (although it may have an initial surge of popularity to begin with). Additionally, users may only rate music they like [12] (as opposed to rating both music they like *and* dislike), hence introducing a positive bias in the ratings. For these two reasons, we must compensate by finding a way to decrease the popularity with time.

The amount by which we decrease popularity was empirically determined by analysing the movement of individual tracks on weekly top 50 charts. The period examined was for the week starting Sunday 4th January 1998 to the week starting Sunday 30th July 2000. This was a total of 135 weeks. The charts used were "The Oz Net Music" charts[1]. The Oz Net Music chart is an independent music chart compiled from a variety of sources.

Only tracks which had been in the charts for longer than sixteen weeks were analysed. Tracks too close to the starting and ending weeks were not used as their entire chart life cycle would not be accurately represented. 103 tracks were plotted. A graph showing mean chart rankings of these tracks over time is displayed in Figure 4.3.

Analysis of this graph would allow us to determine an appropriate function with which to decrease the popularity of tracks with over time.

We must differentiate between *historical* and *contemporary* popularity. The appropriate function referred to above is only suitable for contemporary music. Historical artists, such as the Beatles, are no longer in contemporary music charts, but they are still very popular. A different popularity decline model, which decreases much more slowly would need to be used for historically popular music.

4.6.5 Populating the popularity figures database

We initially give each track a raw popularity figure of 0.5 (refer to Figure 4.4).

By raw, we refer to its numerical rating between 0 and 1 inclusive. It is the unnormalized global rating.

Let us now assume that over time, the popularity figures have changed to the following, as shown in Figure 4.5.

Using these example popularity figures, we will show how local and global popularity are calculated, as well as showing how the popularity of genres is calculated by propagating popularity figures from leafnodes of the musical genre tree up through the tree.

4.6.6 Determining local popularity

Local popularity is the normalized popularity figure within a genre and is determined as follows:

$$pop_L(item) = \frac{\gamma(item)}{\sum_{i=1}^{n} \gamma_i^{genre}}$$

[1] `http://www.q-net.net.au/~methinks/oz_net_music_chart.html`

where pop_L denotes local popularity, $\gamma(item)$ denotes the raw popularity of the particular item (track or artist), and $\sum_{i=1}^{n} \gamma_i^{genre}$ denotes the sum of all the raw popularities within a given genre.

For example, for a given track, say A, classified under the *rock* genre, we can obtain the local popularity within the *rock* genre as:

$$pop_L(A) = \frac{0.8}{\sum_{i=1}^{n} \gamma_i^{rock}} = \frac{0.8}{1.8} = 0.4444$$

Since a track could belong to more than one genre, its local popularity could vary from one genre to another.

4.6.7 Determining global popularity

Global popularity is the normalized popularity over the entire music database and is determined as follows:

$$pop_G(item) = \frac{\gamma(item)}{\sum_{i=1}^{n} \gamma_i^{all\ tracks}}$$

where pop_G denotes global popularity, $\gamma(item)$ denotes the raw popularity of the particular item (track or artist), and $\sum_{i=1}^{n} \gamma_i^{all\ tracks}$ denotes the sum of the raw popularities of all the tracks in the entire database.

For example, the global popularity of track A is:

$$pop_G(A) = \frac{0.8}{\sum_{i=1}^{n} \gamma_i^{all\ tracks}} = \frac{0.8}{5.35} = 0.1495$$

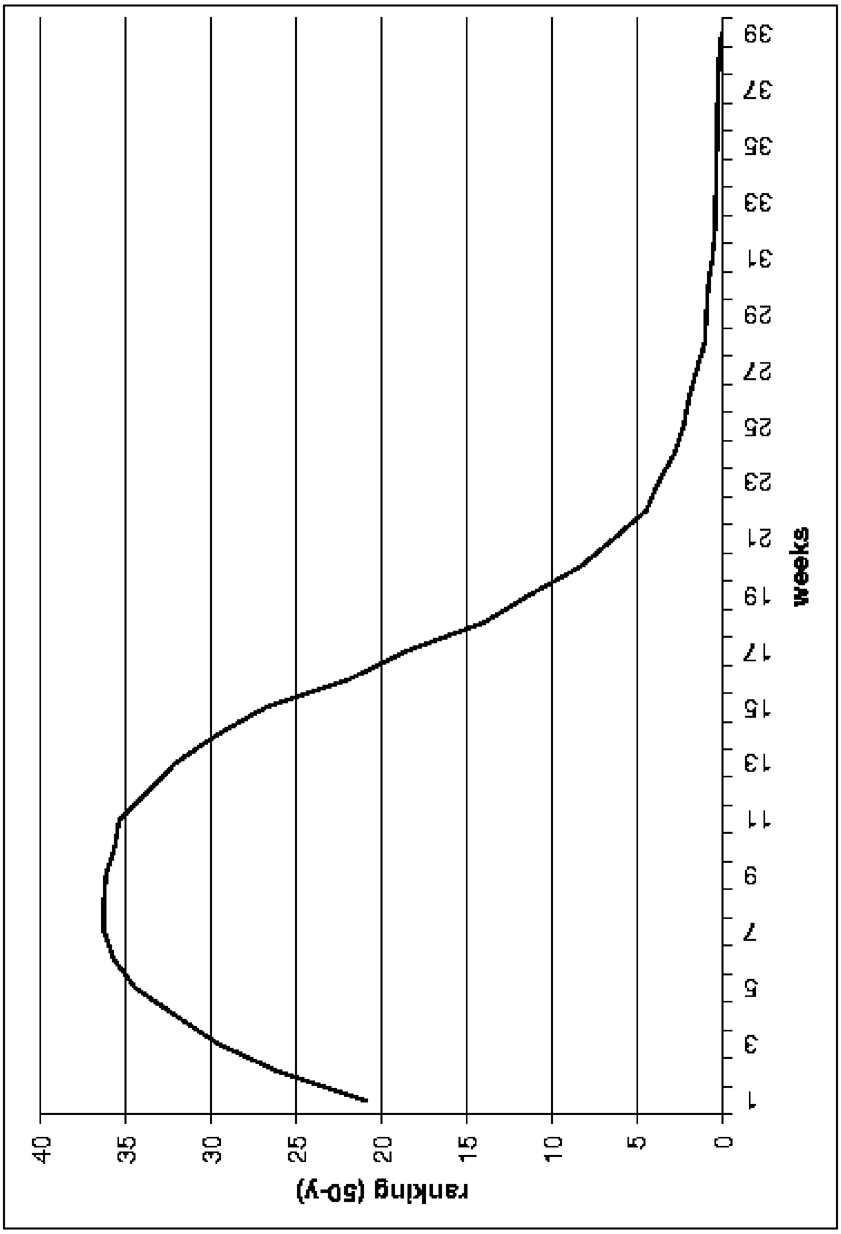


Figure 4.3: Graph of mean chart rankings over time

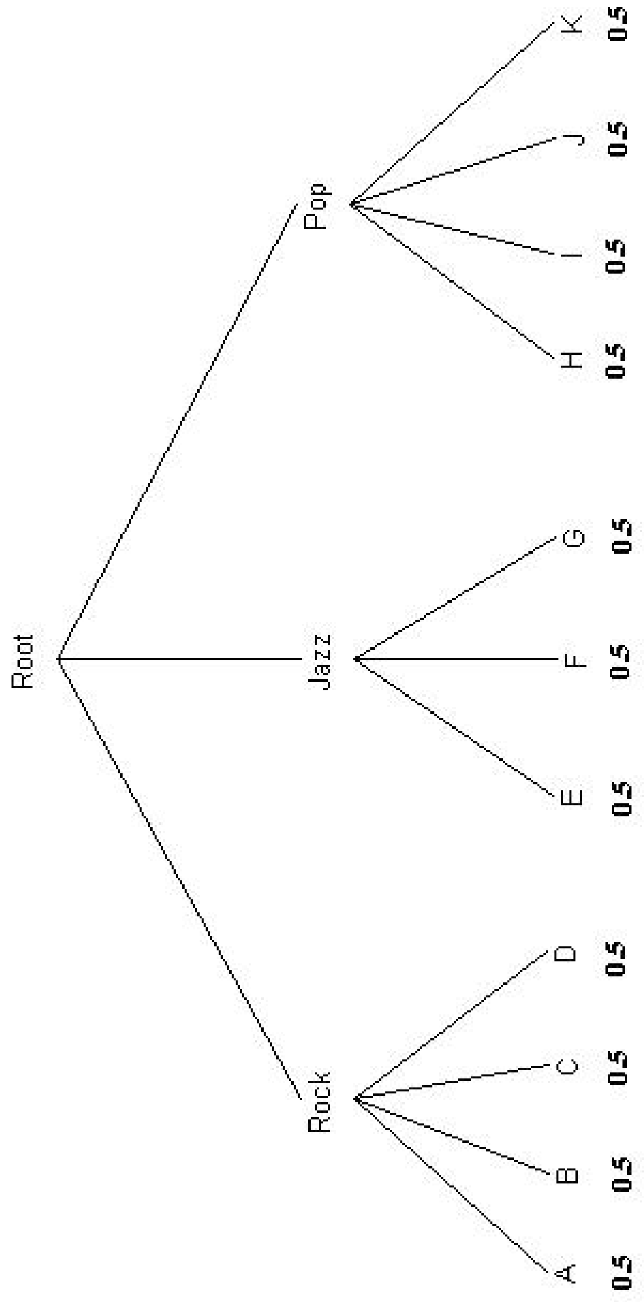


Figure 4.4: Initial popularity figures

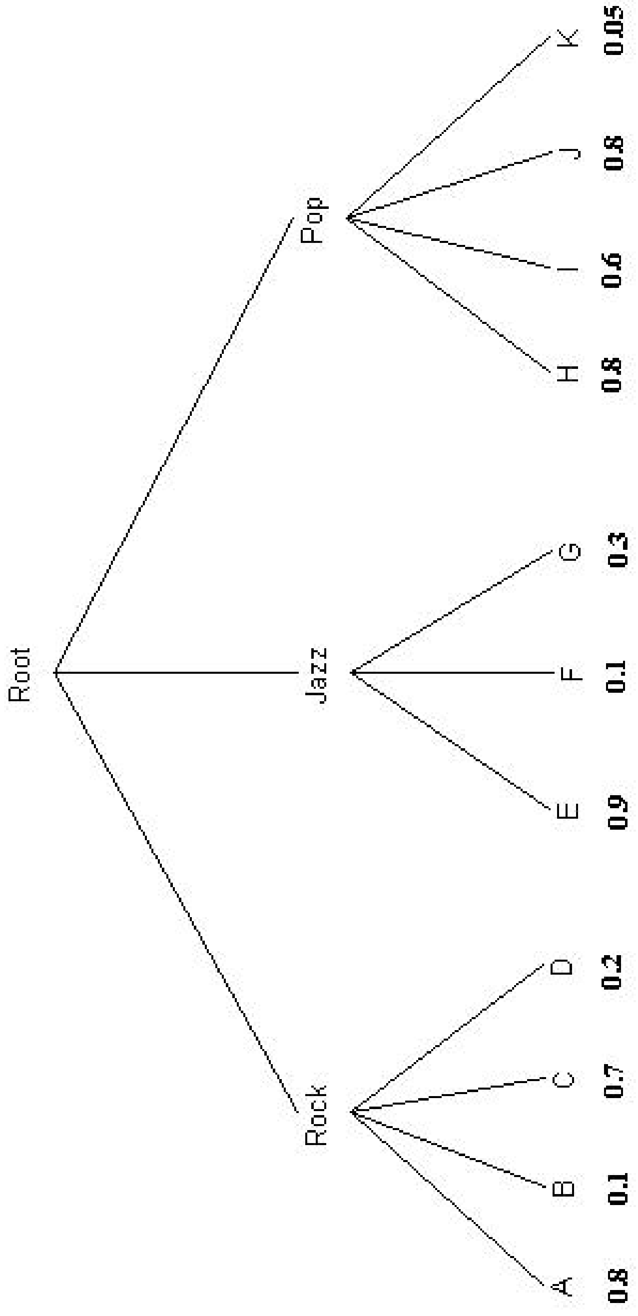


Figure 4.5: Popularity figures change with time

The raw, local and global popularity figures for the tracks in our example are tabulated as follows.

track	raw	local	global
A	0.8	0.4444	0.1495
B	0.1	0.0556	0.0187
C	0.7	0.3889	0.1308
D	0.2	0.1111	0.0374
E	0.9	0.6923	0.1682
F	0.1	0.0769	0.0187
G	0.3	0.2308	0.0561
H	0.8	0.3556	0.1495
I	0.6	0.2667	0.1121
J	0.8	0.3556	0.1495
K	0.05	0.0222	0.0093

We see here that tracks A and H have the same raw popularity but differing local popularities showing that although a track may be popular within its genre, this does not necessarily mean it is popular overall (globally).

Let us now assume track A belongs both to *rock* and *pop*, denoted A_1 and A_2 respectively. Although positioned in two different locations in the musical genre tree, A_1 and A_2 are considered as one track for the purposes of raw popularity and global popularity calculations.

However, they will be considered separately in local popularity calculations, and both A_1 and A_2 are used in determining the popularity of genres by propagation.

4.6.8 Determining genre popularity figures

Genre popularity figures are determined by propagating figures up the tree from the leaf nodes.

We do this by determining the mean raw popularity within each genre, then dividing this by the sum of the mean raw popularity of each genre. This is done with both track and artist popularity figures. We must also consider the popularity rating of genres from user profiles. The final genre popularity is calculated from tracks, artists and user profiles as follows.

$$pop(genre) = \alpha \frac{\overline{\gamma}_{genre}^{track}}{\sum_{i=1}^{n} \overline{\gamma}_{genre\ i}^{track}} + \beta \frac{\overline{\gamma}_{genre}^{artist}}{\sum_{i=1}^{n} \overline{\gamma}_{genre\ i}^{artist}} + \delta \overline{\gamma}_{genre}^{user\ profile}$$

where $\overline{\gamma}_{genre}$ refers to the mean raw popularity within a particular genre and $\sum_{i=1}^{n} \overline{\gamma}_{genre\ i}$ refers to the sum of the mean raw popularities within all the n genres of the entire database. The superscript of ^{track}, ^{artist} or $^{user\ profile}$ denotes popularity from track, artist or user profile respectively. α, β and δ denote the weight given to each factor, as determined by radio station administrators.

For example, the popularity figure based on tracks, of *rock* is given by:

$$pop(rock) = \frac{\overline{\gamma}_{rock}}{\sum_{i=1}^{n} \overline{\gamma}_{genre\ i}} = \frac{0.45}{1.4458} = 0.3112$$

If we ignore user profiles and results from artist popularity figures, this gives us the following updated musical genres tree with popularity figures as shown in Figure 4.6.

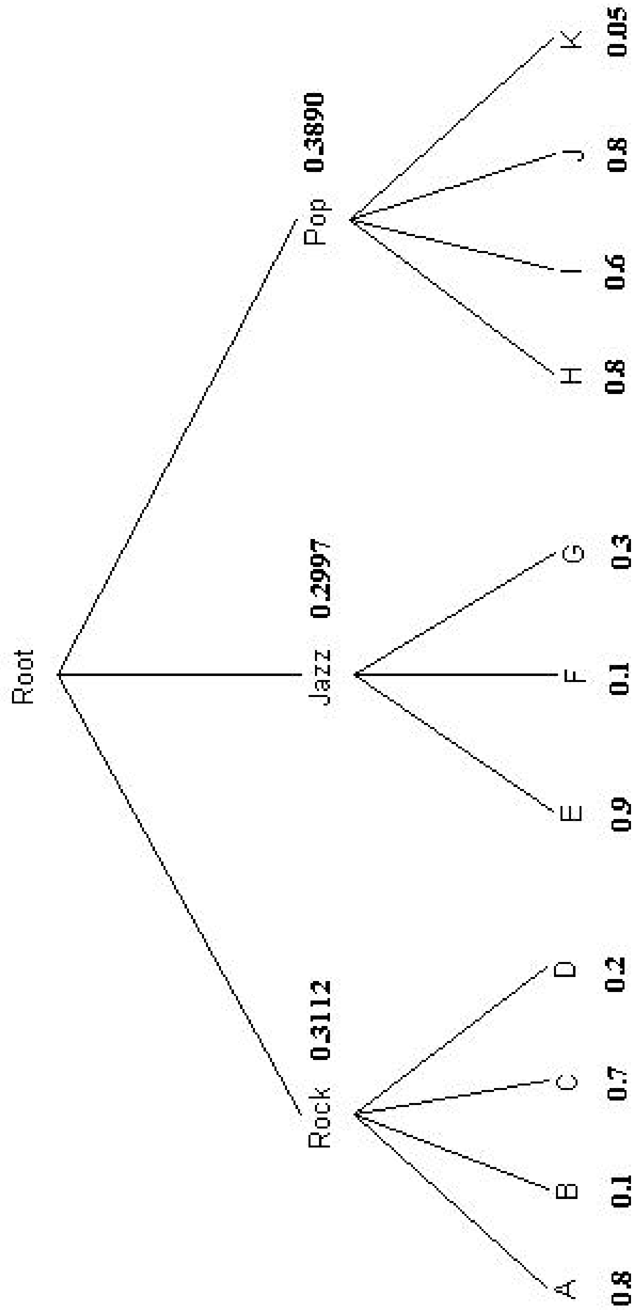


Figure 4.6: Updated musical genres tree

4.7 Popularity–based individual track selection

We present two popularity figures, global, and local. (We also use raw figures which are simply unnormalized global figures).

Global refers to a track's overall popularity within the entire music database.

Local refers to a track's popularity within a given genre. For example, "Billy Joel – Pianoman" may be popular within the "Classic Rock" genre but its overall (global) popularity may be quite low.

We may choose to have an overall top ten countdown of the station's top ten tracks, in which case we would use the global popularity figures, or we may choose to play the most popular tracks from a smattering of different genres, in which case we would use local popularity figures. We may also choose to play some popular tracks from the more obscure genres, in which case both local popularity and popularity of genres would be used.

To select the tracks based on popularity we consider the normalized popularity in question, and arrange the tracks so they mutually exclusively occupy a space somewhere between 0 and 1 inclusive. We then randomly select a number between 0 and 1 inclusive and select the track in which the randomly selected number lies. For example, if the global popularities of the tracks of some hypothetical database are as follows:

Track	Raw popularity
A	0.25
B	0.1
C	0.3
D	0.2
E	0.15

We then arrange these values like so:

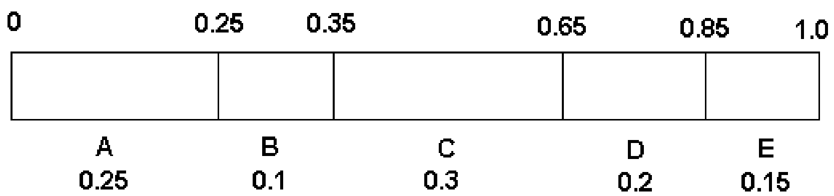


Figure 4.7: Example selection bar

Therefore, if our randomly selected number was 0.67, we would select track D, while if it was 0.28, we would select track B. If the randomly selected number lands on a boundary, we select the track which starts on that boundary, e.g. if our randomly selected number was 0.35, we would select track C.

However, pure popularity–based track selection poses some problems. If we are only playing popular music, how can we play new music to pleasantly surprise existing listeners, and to attract new listeners? Also, how can we cover our catalogue? For these reasons, track selection must be based on more than just popularity.

4.8 "Music programme" generation

4.8.1 Why music programmes?

A programme (e.g., an hour's worth of well organized music tracks) is more natural than 10–15 individually selected tracks. Having a programme ensures that similar tracks are not clustered in the same time period. It also means that transition between different tracks can be made as smoothly as possible based on tempo and transition tracks. With $n!$ possible arrangements of n tracks, the problem of music generation becomes one of combinatorial complexity. Therefore, methods must be put in place to help to reduce the computational requirements of such a task. Additionally with a programme, station administrators have the opportunity to allow users to see what tracks are to be played, hence increasing the comprehensiveness of the station's playlist.

In a large database, it may be quite time consuming to individually select each track, and indeed, it may not be possible to do so in real time. This necessitates the need for music programmes.

4.8.2 Generation of music programmes

In generating music programmes, we must consider the various sources and weightings of each source used in individual track selection. We consider the following sources.

1. Popularity — what users of the station are rating as popular (via user profiles, requests and ratings) (i.e., what they want to hear). Studies in psychology and music theory show that humans have a natural desire for repetition, i.e. listening to music one is already familiar with [7] [8]. Previous discussion has shown how we can compute the popularity of tracks and how we can use it to select the tracks.

2. Catalogue coverage — station administrators may need to cover a certain part of a catalogue (for example, they may have business ties with SONY and have an agreement to play a given percentage of SONY's catalogue).

3. Surprise element — although humans typically expect some repetition, they also like being pleasantly surprised with new music they enjoy but have not yet heard [13]. As well as pleasantly surprising existing listeners, playing new material will hopefully attract new listeners too.

4. Special sources/other factors — this could include supporting a local band by playing some of their music on air. Other factors also include repetition. If a track has been recently played (within some time period determined

by radio station administrators), then it can not be selected. Here, even if the popularity is very high, we can use this consideration to prevent it from being selected too often.

Figure 4.8: Music programme generation weights bar

Sources 1,2 and 3 may be assigned simple static weights (see Figure 4.8) that determine their proportion in the overall sequence. The proportion could be in terms of duration of the tracks or in the number of titles.

For example, if we assign $w_1 = 0.6$, $w_2 = 0.3$ and $w_3 = 0.1$, this would mean that for a given programme, 60% of the tracks would be chosen on the basis of popularity, a 30% of the tracks would be chosen from catalogue coverage, and 10% of the tracks would be the "surprise element". Note that if we only wanted to select a only few tracks, such as four or less, another selection method such as considering the weights as probabilistic rather than solid, or manual source selection by station administrators would need to be considered.

Let us assume we generate a programme of 10 tracks, with A_i, B_i and C_i denoting tracks from sources 1, 2 and 3 respectively. The generated programme could be: $A_1, B_1, B_2, A_2, C_1, A_3, A_4, B_3, A_5, A_6$.

Note that if a track has already been recently played (within some time period determined by radio station administrators), then it can not be selected.

We now introduce special sources.

4.8.3 Special sources/other factors

Special sources refer to such things as supporting local bands or doing a live broadcast of a concert.

A special source may be that we wish to support a local band by playing one of their tracks every six hours. A condition such as this means dynamic weights for this source must be used. For example, if we wish to support the local band in the manner described above (we denote the weight due to this special source as $w_{4,1}$, we may be generating six 60 minute programmes. $w_{4,1}$ will be set to 0 for five of these programmes and will be set to some value such as 0.08 for one of these programmes (to ensure the track from the local band is included in the programme).

However, if we do not know in advance of the special source, it may be necessary to override the other sources to ensure the conditions laid out by the special sources are satisfied. For example, if we are to broadcast a live concert for two hours, we would set an override flag on this special source to ensure that tracks are not selected due to other sources such as popularity, catalogue coverage and surprise element.

If we now revisit our example programme described above, let us assume the programme plays in the time span in which a track from the local band we are supporting must be played. The track denoted D_1, would be inserted into the programme at the appropriate place (to ensure that as close as possible to six hours had elapsed since another track by that local band had been played), and would push the end track off the programme. The precise point of insertion would be influenced by its tempo characteristics and the overall smoothness of the generated sequence (see discussion below).

Our new example programme might be: $A_1, B_1, B_2, A_2, D_1, C_1, A_3, A_4, B_3, A_5$.

Note A_6 has fallen off the end of our programme.

4.8.4 Ensuring a smooth programme

After considering these sources to select the tracks, we must now arrange them in such a way that there is a smooth transition between tracks and a good variety within the programme. For example, in a 60 minute programme, we do not want all rock tracks to be played in the first 20 minutes; we want to spread them out over the entire hour.

"Transition tracks" may be used to help balance our programme. These are tracks which have been classified as belonging to more than one genre (e.g. pop and rock), so they may be used to transition from pop to rock. By doing so, we fix the allowable arrangements of the selected tracks, and hence reduce the candidate sequences.

As well as using transition tracks, the tempo of tracks can also be used. We do not want to transition from a very fast track straight into a very slow track. Instead we do this gradually from fast to medium to slow.

We give each track a "tempo rating" which indicates a track's starting and ending tempos, based on a five point scale, giving a total of 25 possible tempo ratings. We consider both the starting and ending tempo of each track to give us greater control over our tempo transitions and because some tracks do not necessarily start and end at the same tempo. The ratings are shown in the following table.

starting/ending	vs	s	m	f	vf
vs	vsvs	vss	vsm	vsf	vsvf
s	svs	ss	sm	sf	svf
m	mvs	ms	mm	mf	mvf
f	fvs	fs	fm	ff	fvf
vf	vfvs	vfs	vfm	vff	vfvf

Key

Tempo rating	Meaning	Beats per minute range
vs	very slow	< 40 bpm
s	slow	40–60 bpm
m	medium	60–120 bpm
f	fast	120–200 bpm
vf	very fast	> 200 bpm

The tempo rating will help in smoothly transitioning between tracks. For example, a track which ends very slowly may precede a track which starts very slowly, or slowly.

As we have $n!$ possible arrangements, we must reduce the number of possible combinations. Our first step in doing this is to position any transition tracks in a place so as to allow a good balance in the programme. Our second step is to use the tempo to ensure smooth transitions (this rules out such things as placing a track which starts very fast directly after a track which ends very slowly).

Once a sequence has been generated from this reduced set of possible arrangements, its "smoothness rating" is calculated, which is a measure of how "smooth" a sequence is, in other words, how seamless the transitions between individual tracks of the sequence are, and how balanced the sequence is in terms of genre and tempo distributions (i.e., are all "rock" tracks clustered in the same time frame; are all fast tempo tracks clustered in the same time frame? — we want to avoid this.).

If the smoothness rating of the sequence is below some acceptable threshold (as determined by station administrators), then a new arrangement from the reduced set of possible arrangements is tried. The algorithm will eventually stop after a certain number of attempts and take the best result so far, if failure to obtain an arrangement below the desired threshold has occurred.

Failing the use of transition tracks or tempo to ensure a smooth transition, we use a fade in/fade out effect to smooth the transition between tracks.

Calculating the smoothness rating

Each transition between two tracks has an associated rating. The sum of these ratings gives the overall smoothness rating. The ratings are tabulated as follows.

Transition type	Associated rating
Transition track (0 degrees of difference)	0.75
Transition track (1 degree)	0.7
Transition track (2 degrees)	0.65
Transition track (>2 degrees)	0.6
Tempo (0 degrees)	0.5
Tempo (1 degree)	0.4
Tempo (2 degrees)	0.3
Fade in/out effect	0.25

The higher the smoothness rating of a programme, the better. Degrees of difference between adjacent tracks refers to how many different tempos are transitions from one track's ending tempo to the next track's starting tempo. For example, slow to slow is 0 degrees, slow to medium is 1 degree, and slow to fast is 2 degrees. If more than 2 degrees would be necessary and no transitions are available then a fade in/out effect is used.

4.8.5 Selecting "surprise element" tracks

"Surprise element" tracks refer to tracks which a station's listeners haven't heard, but will be pleasantly surprised by. "Surprise element" tracks also help in attracting new listeners to the station. "Surprise element" tracks can either be new music or older undiscovered music. New music can only be manually added to the station and flagged as potential "surprise element" music. For other music, we will use the artist popularity to select potential "surprise element" tracks. This is achieved by playing little heard before music from popular artists.

Chapter 5

Simulation/Results

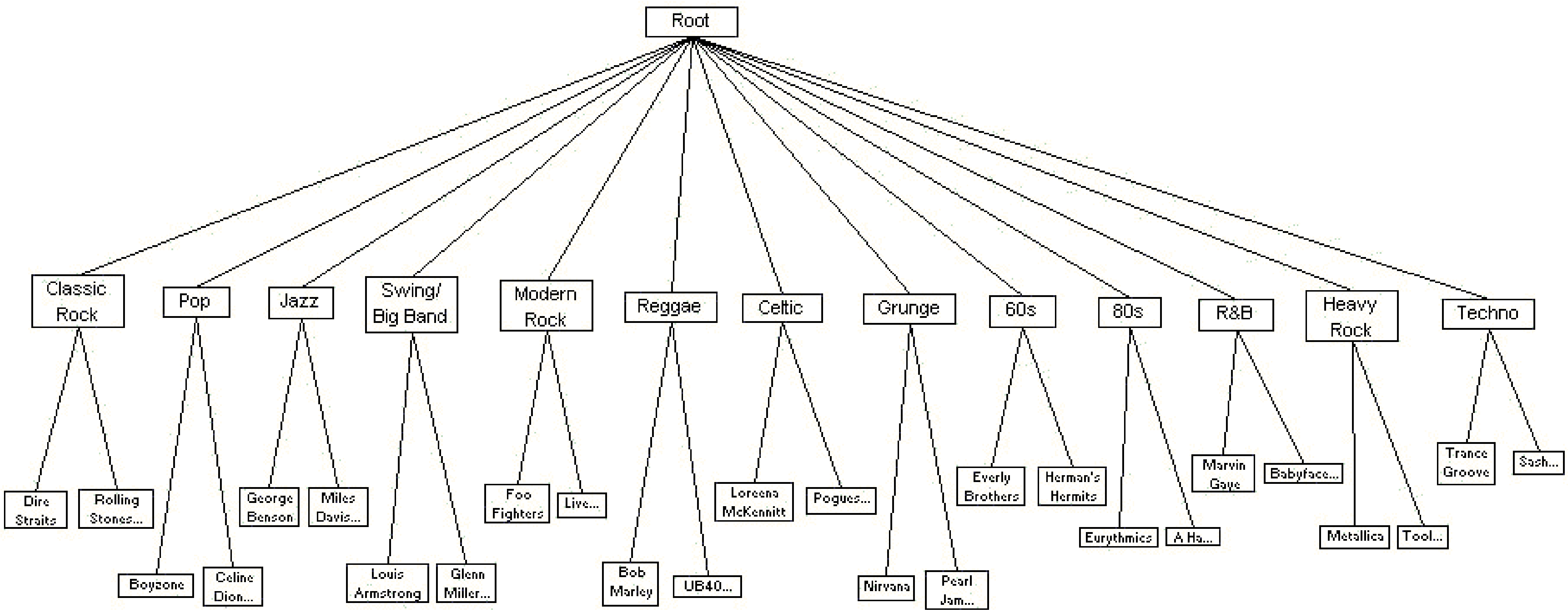
5.1 Overview

A simulation[1] of the work described in this report was implemented in C. 452 tracks from 115 unique artists covering 13 genres were used. Each track's starting and ending tempos were manually obtained. Each artist was manually classified into appropriate genre(s). We assumed that a track's genre(s) was the same as its artist (although this many not strictly be the case). Tracks classified under more than one genre are considered to be transition tracks.

The initial popularities of all items were set to zero. We assumed our simulated radio station had 100 users.

A musical genre tree of the simulated station showing all genres and a subset of artists from each genre can be seen by referring to Figure 5.1.

[1] Although a working radio station has been implemented (http://132.181.8.2), the ideas proposed in this report have not been implemented on this station. The data and the source code for the simulation is available on request.


Root
Classic Rock
Pop
Jazz
Swing/ Big Band
Modern Rock
Reggae
Celtic
Grunge
60s
80s
R&B
Heavy Rock
Techno
Dire Straits
Rolling Stones...
Boyzone
Celine Dion...
George Benson
Miles Davis...
Louis Armstrong
Glenn Miller...
Foo Fighters
Live...
Bob Marley
UB40...
Loreena McKennitt
Pogues...
Nirvana
Pearl Jam...
Everly Brothers
Herman's Hermits
Eurythmics
A Ha...
Marvin Gaye
Babyface...
Metallica
Tool...
Trance Groove
Sash...

5.2 Popularity

User profiles, user requests and user online ratings were used to determine popularity of track and artist. The genre popularity was then determined by propagating values from the leaf nodes of the track and artist musical genre trees up the tree, as described previously. We assumed no external sources were in action during our simulation.

5.2.1 User profiles

Only artists and genres were implemented. [12] reports that a user's mean rating of items in the profile on the seven point scale used was 3.7. We attempted to obtain a similar mean in our simulation by using the following probabilities of selection per rating option.

Rating option	Probability
1	0.22
2	0.20
3	0.14
4	0.03
5	0.06
6	0.19
7	0.16

Note the higher probabilities of selection for the very low and very high rating options which reflects human selection patterns. Users rated 35 core artists and a remaining 30 were randomly selected from the remaining artist list. Users rated all 13 genres. We assumed there was a 25% chance the user was unfamiliar with a given artist, and would therefore not rate that given artist. We assumed 100% familiarity with genres.

A sample user profile may be found in Appendix B.

5.2.2 User requests

We assumed that 200 requests were made per day. Each track that was requested influenced the popularities of both that track and that track's artist.

In a real-life situation, the more popular tracks are going to receive more requests. To more closely emulate this, we assigned tracks into three popularity bands (high popularity, medium popularity and low popularity). 40% of the music was placed into the high popularity band, 40% into the medium popularity band and 20% into the low popularity band. Tracks in the high popularity band had a 0.6 probability of selection, tracks in the medium popularity band had a 0.3 probability of selection, while tracks in the low popularity band had a 0.1 probability of selection.

5.2.3 User online ratings

We assumed that every third track would be rated by 25 users. The following probabilities of selection per rating option were used.

Rating option	Probability
1	0.23
2	0.20
3	0.14
4	0.20
5	0.23

As with user requests, a track's rating influences both the track and the track's artist popularity.

5.3 Calculating popularity

All examples in this section will use data obtained from a simulation simulating ten days of station uptime.

5.3.1 Artists

Artist popularity was calculated from the user profiles, user requests and user online ratings with respective weights of 0.2, 0.4 and 0.4.

The top ten artists were as follows.

Rank	Artist	Global popularity
1	Van Morrison	0.010241
2	Robbie Williams	0.010123
3	Live	0.010103
4	Brian Poole & The Tremeloes	0.010083
5	Duane Eddy	0.009903
6	Celine Dion	0.009818
7	The Dave Brubeck Quartet	0.009728
8	Irene Cara	0.009701
9	The Corrs	0.009661
10	Bob Dylan	0.009634

5.3.2 Tracks

Track popularity was calculated from user requests and user online ratings with respective weights of 0.7 and 0.3 The user requests had a higher weight because the raw numbers obtained from the user requests were much smaller than those obtained from the user online ratings.

The top ten tracks were as follows. (Tracks are denoted in Artist – Track format).

Rank	Track	Global popularity
1	Rolling Stones – Angie	0.003403
2	Crowded House – When You Come	0.003032
3	Tom Petty & The Heartbreakers – Breakdown	0.002992
4	Bob Dylan – Like A Rolling Stone	0.002986
5	Berlinda Carlisle – Heaven Is A Place On Earth	0.002967
6	Five – Got The Feelin'	0.002939
7	Bob Dylan – Mr. Tambourine Man	0.002936
8	Mariah Carey – Dreamlover	0.002933
9	Van Morrison – Baby Please Don't Go	0.002903
10	Eric Clapton – Forever Man	0.002886

5.3.3 Genres

This was calculated from artist popularity, track popularity and user profiles with respective weights of 0.45, 0.45 and 0.1.

The genre popularities are as follows.

Rank	Genre	Genre popularity
1	Pop	0.277913
2	Classic Rock	0.239286
3	Modern Rock	0.217851
4	80s	0.117094
5	60s	0.115470
6	Jazz	0.090598
7	Swing/Big Band	0.087598
8	Heavy Rock	0.069936
9	R&B	0.064456
10	Grunge	0.060269
11	Techno	0.060094
12	Celtic	0.057231
13	Reggae	0.056868

5.4 Popularity–based track selection

We discuss some case scenarios.

1. Top n overall. This is based purely on global track popularity. For an example of top n overall track selection, please see the above Tracks section.

2. Top n within a genre. This is based purely on local track popularity. For example, the ranked list of all tracks in the "techno" genre are as follows, with the first entry denoting the most popular track and the last entry the least.

Rank	Track	Local popularity
1	Sash – Encore Une Fois	0.075473
2	Trance Groove – Ange Gardien	0.075321
3	Trance Groove – Morning Zoo	0.073111
4	Trance Groove – Stone Soup	0.072288
5	Anabolic Frolic – Eternity	0.071876
6	Anabolic Frolic – Crowd Control	0.071811
7	Trance Groove – Dschang Thang	0.071337
8	Sash – Mighty Break	0.068084
9	Trance Groove – Trainspotting	0.067542
10	Sash – It's My Life	0.065765
11	Sash – Ecuador	0.065094
12	Sash – The Final Pizzi	0.063858
13	Anabolic Frolic – Killer	0.059525
14	Anabolic Frolic – You're Mine	0.051767
15	Anabolic Frolic – Eternity (2)	0.047108

Note the the sum of the local track popularities is one, as is the case with local popularities.

3. "Surprise element" selection. This is based on selecting popular tracks from the more obscure genres. Tracks belonging to genres with low popularity are less likely to be selected, however, this does not mean that tracks belonging to these genres are bad. For this reason, we select the more popular tracks from these genres.

   Bearing in mind the genre popularities displayed earlier, the following is a selection of the top 2 tracks from the five more obscure (least popular) genres. This selection was obtained by combining both local artist and genre popularity.

Artist	Genre
Bob Marley - Exodus	Reggae
UB40 - Food For Thought	Reggae
Levellers - Hope St	Celtic
Loreena McKennitt - Prologue	Celtic
Sash - Encore Une Fois	Techno
Trance Groove - Ange Gardien	Techno
Green Day - Chump	Grunge
Green Day - Welcome To Paradise	Grunge
Marvin Gaye - Ain't No Mountain High Enough	R&B
Marvin Gaye - Sexual Healing	R&B

4. Another "surprise element" selection technique is to select obscure tracks from popular artists. This combines global artist popularity with global track popularity. An example selection based on the top ten artists is as follows.

Track	Artist rank
Van Morrison – Bright Side Of The Road	1
Robbie Williams – Angels	2
Live – I Alone	3
Brian Poole & The Tremeloes – Twist & Shout	4
Duane Eddy – Rebel Rouser	5
Celine Dion – Beauty And The Beast	6
The Dave Brubeck Quartet – Strange Meadow Lark	7
Irene Cara - Fame	8
The Corrs - Forgiven Not Forgotten	9
Bob Dylan - The Times They Are A-Changin'	10

5.5 Arranging the tracks into a programme

We consider the following ten randomly selected tracks.

Track	Start	End	Genre(s)
Boston – Walkin' At Night	1	1	Classic Rock, Modern Rock, Pop
Foo Fighters – Monkey Wrench	4	3	Modern Rock, Classic Rock
Boyzone – And I	3	1	Pop
Nirvana – Intro	2	3	Grunge, Modern Rock
Phil Collins – Another Day In Paradise	1	2	Classic Rock, Modern Rock, Pop
Bob Dylan – Don't Think Twice It's All Right	3	2	Classic Rock, Modern Rock
Babyface – I Said I Love You	1	2	R&B, Pop
Benny Goodman – Goody, Goody	3	2	Swing/Big Band, Jazz
Boston - Livin' For You	2	3	Classic Rock, Modern Rock, Pop
The Beach Boys - The Warmth Of The Sun	3	2	60s, Classic Rock, Pop

Where *Start* and *End* refer to the starting and ending tempos of each track with 1 denoting very slow through to 5 denoting very fast.

We calculate the smoothness rating of this selection.

Transition	Tempo difference	Genre overlap	Transition rating
1–2	3	$\surd$	0.6
2–3	0	$\times$	0.5
3–4	1	$\times$	0.4
4–5	2	$\surd$	0.65
5–6	1	$\surd$	0.7
6–7	1	$\times$	0.4
7–8	1	$\times$	0.4
8–9	0	$\times$	0.5
9–10	0	$\surd$	0.75

The smoothness rating of the above selection totals 4.9.

We now attempt to increase the smoothness rating of the above selection by using the following steps.

1. We evenly distribute the transition tracks belonging to 3 genres throughout the programme

2. We now traverse the list starting from the head and insert the track which will achieve the lowest tempo difference.

A new order is obtained.

Track	Start	End	Genre(s)
Boyzone – And I	3	1	Pop
Boston – Walkin' At Night	1	1	Classic Rock, Modern Rock, Pop
Bob Dylan – Don't Think Twice It's All Right	3	2	Classic Rock, Modern Rock
Phil Collins – Another Day In Paradise	1	2	Classic Rock, Modern Rock, Pop
Babyface – I Said I Love You	1	2	R&B, Pop
Boston - Livin' For You	2	3	Classic Rock, Modern Rock, Pop
Benny Goodman – Goody, Goody	3	2	Swing/Big Band, Jazz
The Beach Boys - The Warmth Of The Sun	3	2	60s, Classic Rock, Pop
Nirvana – Intro	2	3	Grunge, Modern Rock
Foo Fighters – Monkey Wrench	4	3	Modern Rock, Classic Rock

Where tracks in bold denote results of operations pertaining to step one. We calculate the new smoothness rating for this new order.

Transition	Tempo difference	Genre overlap	Transition rating
1–2	0	$\surd$	0.75
2–3	2	$\times$	0.65
3–4	1	$\times$	0.7
4–5	1	$\surd$	0.7
5–6	0	$\surd$	0.75
6–7	0	$\times$	0.5
7–8	1	$\times$	0.4
8–9	0	$\times$	0.5
9–10	1	$\surd$	0.7

Following the steps outlined above, a higher smoothness rating of 5.65 was obtained.

Chapter 6

Conclusion

A comparative study of internet radio stations was performed and results of the study were presented. From this study, we examined and outlined the problem of music selection and music programme generation. We proposed an approach based on popularity, catalogue coverage and the "surprise element".

Methods for obtaining popularity of artists, tracks, and genres, based on sources such as user profiles, user requests and user online ratings were proposed. A method for obtaining genre popularities based on musical genre trees was outlined. Once the tracks have been selected, a method for arranging these tracks and reducing the computational complexity of this operation is proposed. A simulation of an internet radio station was performed and examples from the simulation were presented, as well as an example track arrangement operation.

Although a framework has now been put in place by the work in this report, there is much scope for future work.

A more detailed study of popularity and its variation needs to be conducted. This could involve analysis of data gathered from a real–life (i.e non–simulated) radio station. The user request popularity method needs to be improved as under the current system, popularity figures obtained from this source are too small to be of much use. Classification methods could be examined. This includes formalizing tempo classification and genre classification. For example, tracks/artists belonging to several genres could have normalized weights on each genre they belong to. This would mean that if a track falls under both the rock and pop umbrellas, but is more rock than pop, it could be given a higher weight for rock.

Methods for determining the "smoothness rating" need to be formalized. as well as work on reducing the computational complexity of the track arrangement operations and track selection process (perhaps by tree pruning). Using a genre graph instead of a genre tree could be beneficial and could be investigated.

This is the end of the report. Now go and listen to some good music at http://132.181.8.3:8000.

Acknowledgements

I would like to thank Dr. Donald A. Adjeroh, my supervisor, for his help and patience throughout the year. I would also like to thank Jane McKenzie for proof reading and correcting the writing style of this report and Jackie Waylen, the music librarian of the University of Canterbury for the help she provided.

References

[1] Baldwin, R. **MP3: Dividing Communities**, *IEEE MultiMedia* (1999), April–June, pp. 6–7.

[2] Chen, J. and Chen, A. **Query by Rhythm: An Approach for Song Retrieval in Music Databases**, *Proceedings Eighth International Workshop on Research Issues in Data Engineering, Continuous-Media Databases and Applications*, (1998), pp. 139–146.

[3] Foote, J. **An Overview of Audio Information Retrieval**, *Multimedia Systems* (1999), Vol. 7, No. 1, pp. 2–10, January.

[4] Likert, R. **A Technique for the Measurement of Attitudes**, *Archives of Psychology* (1932), Vol. 140, June.

[5] Loeb, S. **Architecting Personalized Delivery of Multimedia Information**, *Communications of the ACM* (1992), December Vol. 35, No. 12, pp. 39–48.

[6] McNab, R., Smith, L., Witten, I., Henderson, C and Cunningham, S. **Towards the Digital Music Library: Tune Retrieval from Acoustic Input**, *Proceedings of the 1st ACM International Conference on Digital Libraries* (1996).

[7] Meyer, L. **Emotions and Meaning in Music**, University of Chicago Press, Chicago, (1956).

[8] Narmour, E. **The Analysis and Cognition of Melodic Complexity**, University of Chicago Press, Chicago, (1992).

[9] Pachet, F., Roy, P., and Cazaly, D. **A Combinatorial Approach to Content-Based Music Selection**, *IEEE MultiMedia* (2000), January–March.

[10] Rosenthal, R. and Rosnow, R. **Essentials of Behavioral Research: Methods and Data and Analysis**, McGraw Hill, second edition, (1991).

[11] Scheirer, E. **Tempo and Beat Analysis of Acoustic Musical Signals**, *Journal of the Acoustical Society of America* (1998), Vol. 103, No. 1, pp. 588-601.

[12] Shardanand, U. and Maes, P. **Social Information Filtering: Algorithms for Automating "Word of Mouth"**, *CHI* Proceedings (1995).

[13] Smith, D. and Melara, R. **Aesthetic Preference and Syntactic Prototypicality in Music: 'Tis the Gift to Be Simple**, *Cognition* (1990), Vol. 34, pp. 279–298.

[14] Soltau, H., Schultz, T., Westphal, M. and Waibel, A. **Recognition of Music Types**, *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing* (1998).

[15] Zhang, T., Kuo, J. **Content-Based Classification and Retrieval of Audio**, *Proceedings of the SPIE - The International Society for Optical Engineering* (1998), Vol. 3461, pp. 432–443.

Appendix A

Raw survey results

Name	Streaming method	Playlist support	Number of streams	Multiple bandwidth support	Request mechanism	Type
LouisianaRadio.com	ra,mp3	s	1	n	n	l
MEDIAmazing.com	asx	n	30	n	s	p
ChristianRock.Net	ra	m	1	n	m	l
KINY Radio	ra	n	1	n	n	l
WKKY Country 104.7 FM	ra	n	1	n	n	l
RadioMojo.com	mp3	n	1	y	s	p
Beats.dk	mp3	n	3	n	n	l
Flames Radio	asx	n	1	n	n	l
Beethoven.com	ra,asx	n	1	y	n	l
Artlibitum	mp3	a	1	n	n	p
OzWorld	ra,asx	m	1	n	n	l
netFM	asx	n	1	n	n	l
3WK Underground Radio	ra	n	1	n	s	l
Axis Studios - Radio Station	mp3	m	1	n	n	p
BoomBox Radio	ra	n	60	n	n	l
CyberRadio2000	asx	n	40	n	s	l
Digital One	asx	n	6	n	s	l
eJaVe	ra	n	5	n	n	l
erika.net internet radio	mp3	m	1	n	n	p
EyeQRadio	ra, mp3	n	1	n	n	l
radio ya	mp3	m	6	n	n	p
Radi01	ra	n	4	n	n	l
GUITARadio	qt4	s	11	n	n	l
HALO Radio	ra	n	1	y	s	l
Internet Student Radio	ra	n	5	n	n	l
JamRadio	ra	n	1	n	n	l
Dark City Radio	qt4	n	1	n	n	p
Hardradio.com	qt4, asx, ra	n	1	y	n	l
Hacked-Up Radio	mp3	m	1	n	s	p

Name	Streaming method	Playlist support	Number of streams	Multiple bandwidth support	Request mechanism	Type
Music Choice	asx	n	10	n	n	l
Radio Anarchy	ra	n	1	n	n	l
All That Jazz	ra, asx	n	1	y	n	l
Carpedata	mp3	n	1	n	n	l
2000radio.com	asx	n	1	y	n	p
gogaga Internet Radio	ra	m	24	n	s	l
Progged Radio	mp3	m	1	y	m	p
PureHardRock.com	ra, asx	n	1	n	n	l
Radio Hairball	ra	n	1	n	n	l
Capital Gold	ra	n	1	n	n	l
pump100.com	asx	n	1	y	n	l
Radio Fantastica	mp3	n	1	y	n	l
Nerve Radio	ra	n	2	n	n	l
RadioWoodStock	ra	n	1	n	n	l
Smiling Radio	mp3	n	1	n	n	l
The Rave Network	ra	n	1	n	n	l
The Womb	ra	n	1	n	n	l
Internet Radio Hawaii	qt4,asx,ra	n	1	n	n	l
KONG Radio	ra	n	1	n	n	l
BluesBoysMusic.com	ra,asx	m	1	n	n	l
Rap3000	ra	n	1	n	n	l
Angel Radio	ra	n	1	n	n	l

Appendix B

Sample user profile from simulation

```
Sample user profile for COSC460 simulation.

Core artists
------------

Babyface - 7
Beatles - 2
Bee Gees - 6
Bob Dylan - 2
Bob Marley - 2
Bon Jovi - 6
Boyzone - 4
Celine Dion - 6
Collective Soul - 6
Crowded House - 2
Dire Straits - 7
Doobie Brothers - 6
Duke Ellington - 7
Eagles - 4
Ella Fitzgerald - 5
Eric Clapton - 7
Everly Brothers - 7
Loreena McKennitt - 1
Mariah Carey - 1
Metallica - 2
Queen - 7
Robbie Williams - 6
Rolling Stones - 1
Spice Girls - 1
The Beach Boys - 2
The Police - 1
Tom Petty & The Heartbreakers - 6
```

```
Other (randomly selected) artists
---------------------------------

Blink 182 - 6
Bobby Day - 7
Bryan Ferry - 3
Dragon - 6
Fine Young Cannibals - 5
Foreigner - 1
Inner Circle - 7
Irene Cara - 7
Jimmy Ruffin - 1
Keith Sweat - 2
New Order - 7
Nirvana - 1
Pearl Jam - 1
Rod Stewart - 6
Second Coming - 2
The Bangles - 3
The Foundations - 6
TLC - 1
ZZ Top - 6

Genres
------

Classic Rock - 7
Pop - 3
Jazz - 7
Reggae - 3
Swing/Big Band - 1
Modern Rock - 5
Celtic - 7
Grunge - 7
60s - 2
80s - 7
Heavy Rock - 1
R&B - 4
Techno - 6
```