Finding a path through the Juke Box
The Playlist Tutorial

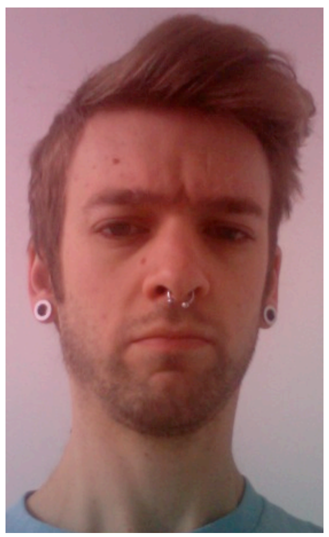
Ben Fields, Paul Lamere

ISMIR 2010

“I still maintain that music is the best way of getting the self-expression job done.”

Nick Hornby

Speakers



Ben Fields
Goldsmiths University of London
benfields.net

Paul Lamere
The Echo Nest
MusicMachinery.com

Goals

- Understand where and why playlists are important
- Understand current and past methods of playlist construction
- Understand the whys and hows of various evaluation methods

Introduction

What is a playlist?

- mixtape
- prerecorded DJ set/mix CD
- live DJ set (typically *mixed*)
- radioshow logs
- an album
- functional music (eg. Muzak)
- any ordered list of songs?

What is a playlist?

We define a playlist as a set of **songs** meant to be **listened** to as a group, usually with an explicit **order**

Why is playlisting important?

- Ultimately, music is consumed through listening
- An awareness of this act of listening is critical to successful MIR application
- The playlist is a formalization of this listening process
- Playlists have a traditional revenue model for artists and labels (e.g. radio)

Brief History of Playlists

Mixed Concert Programs

- Marks the beginnings international combinations of music from multiple composers
- Begins circa 1850 in London
- The idea of a set of music being curated begins to form

Early Broadcast Media

- Moving the ethos of the earlier period onto the radio
- Biggest changes are technology
  - Broadcast = larger simultaneous audience
  - Phonograph brings recorded music
- Initial broadcasts (eg. 1906 - Fessenden) as publicity stunts
- First continuous broadcast 1920 - Frank Conrad

Rock On the Radio

- Radio as a medium begins to push certain genres, especially rock and roll and r ‘n’ b
- Playlist first used to describe (unordered) sets of songs
- Personality driven
  - John Peel
  - Casey Kasem

Disco & Hip-Hop
emergence of the club DJ

- DJs at disco nightclubs, with a mixer and two turntables, saw the birth of the idea of **continuous mixing**
- DJs wanted dancers to not notice song transitions, and techniques such as **beat matching** and **phrase alignment** were pioneered
- Hip-Hop saw this idea pushed further, as DJs became live remixers, turning the turntable into an instrument
- At the same time, club DJs started to become the top billing over live acts, the curator becoming more of a draw than the artist

Last Night A DJ Saved My Life; The history of the disc jockey
Bill Brewster and Frank Broughton

The Playlist Goes Personal

- The emergence of portable audio devices drives the popularity of cassette tapes
- This in turn leads to reordering and combining of disparate material into *mixtapes*
- Mixtapes themselves are traded and distributed socially, providing a means for recommendation and discovery
- In hip-hop, mixtapes served as the first recordings of new DJs featuring novel mixes and leading to current phenomenon of Mix [CD|set|tape] (now on CD or other digital media)

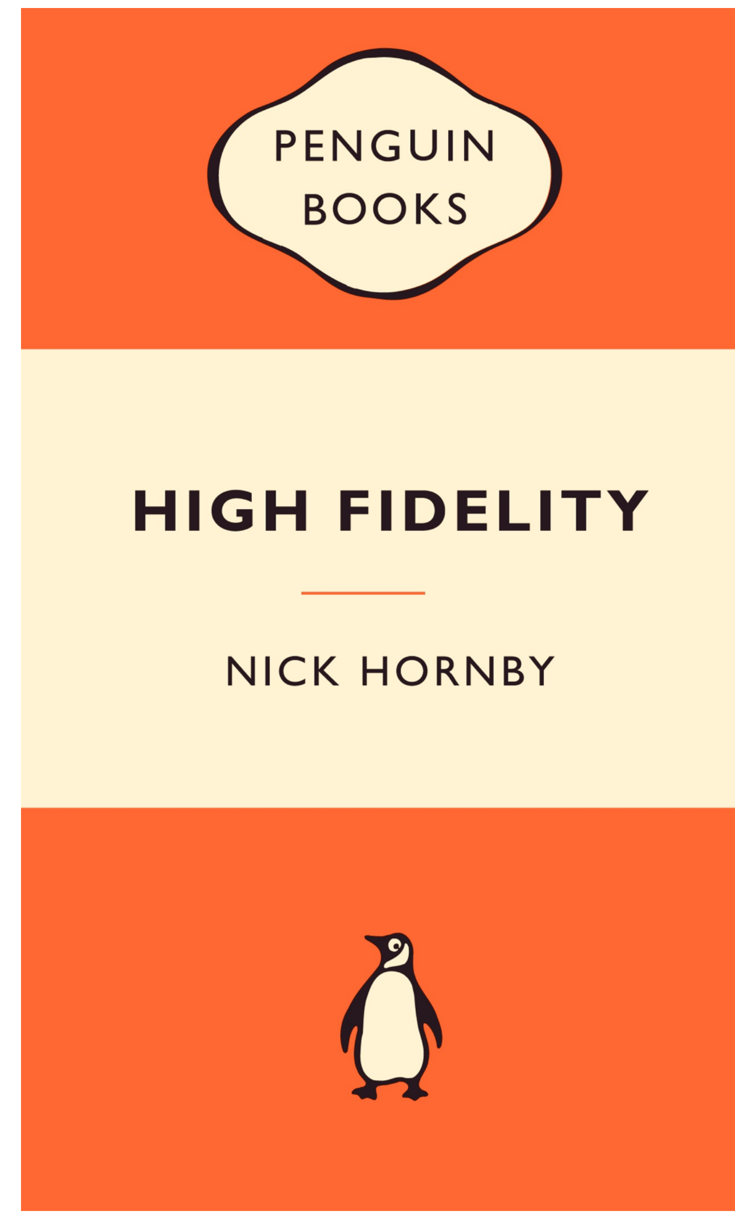
Now With Internet

- The Web's increase in popularity and MP3 audio compression allow for practical sharing of music of the Internet
- This brings the mixtape for physical sharing to non-place sharing.
- Streaming-over-internet radio emerges
- Playlists on the cloud: play.me, spotify, etc.

Aspects of a good playlist

Aspects of a good Playlist

To me, making a tape is like writing a letter — there's a lot of erasing and rethinking and starting again. A good compilation tape, like breaking up, is hard to do. You've got to kick off with a corker, to hold the attention (...), and then you've got to up it a notch, or cool it a notch, and you can't have white music and black music together, unless the white music sounds like black music, and you can't have two tracks by the same artist side by side, unless you've done the whole thing in pairs and...oh, there are loads of rules. - Nick Hornby, *High Fidelity*



Factors affecting a good playlist

- The **songs** in the playlist
  - Listener’s **preference** for the songs
  - Listener’s **familiarity** with the songs
  - Song **coherence**
  - Artist / Song **variety**
  - And more: **freshness**, **coolness**,
- The **order** of the songs:
  - The song **transitions**
  - Overall playlist **structure**
  - **Serendipity**
- The **context**

Factors affecting a good playlist

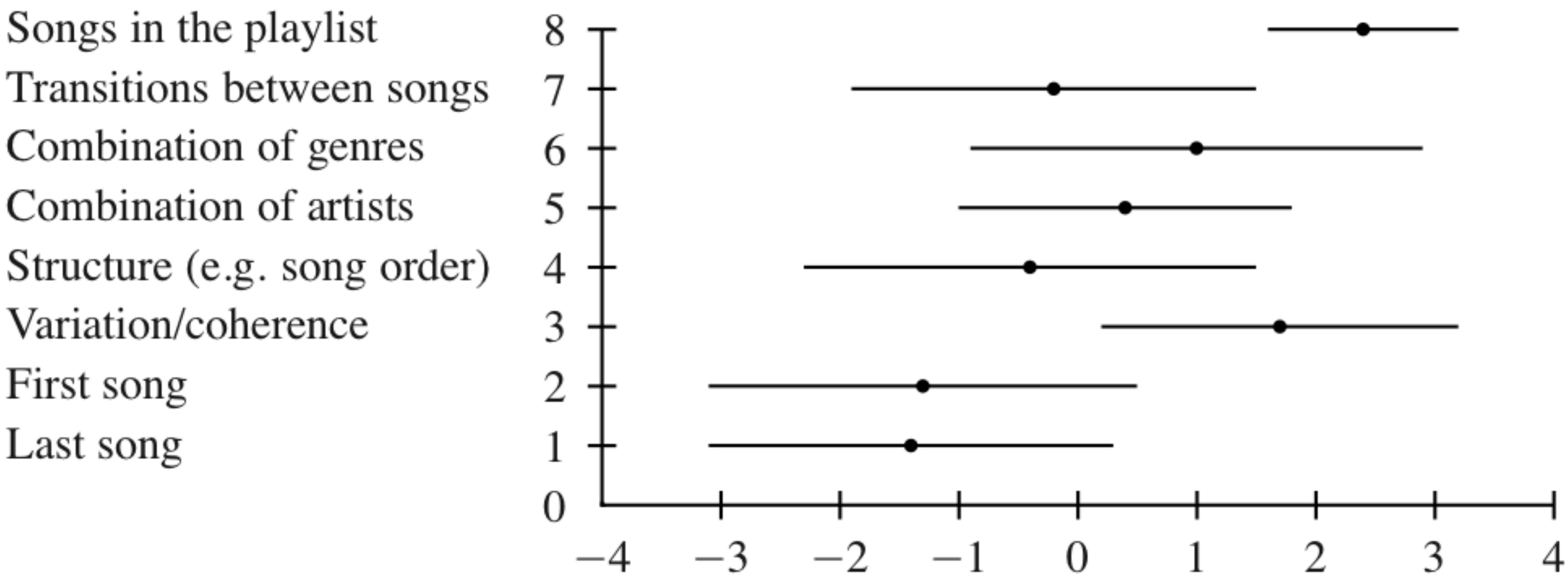


Figure 1: Importance of various factors in creating a playlist.

Survey with 14 participants

Factors affecting a good playlist

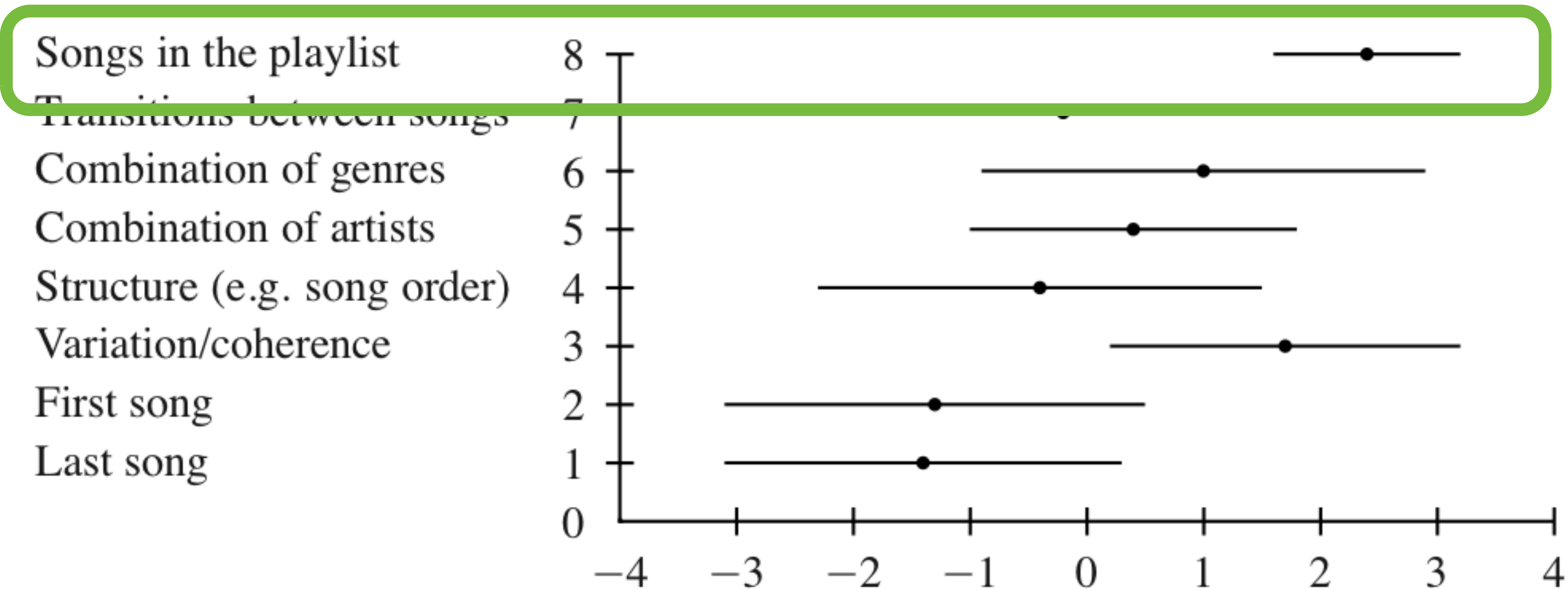


Figure 1: Importance of various factors in creating a playlist.

Survey with 14 participants

Factors affecting a good playlist

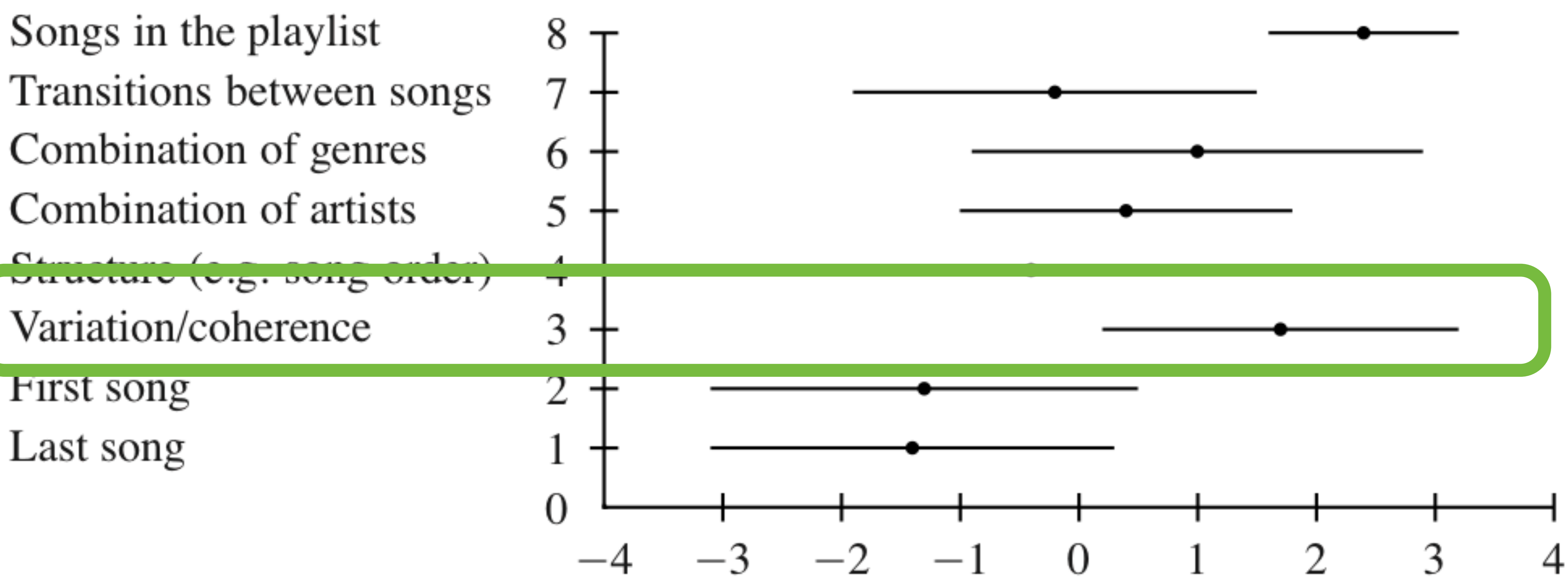


Figure 1: Importance of various factors in creating a playlist.

Survey with 14 participants

Factors affecting a good playlist

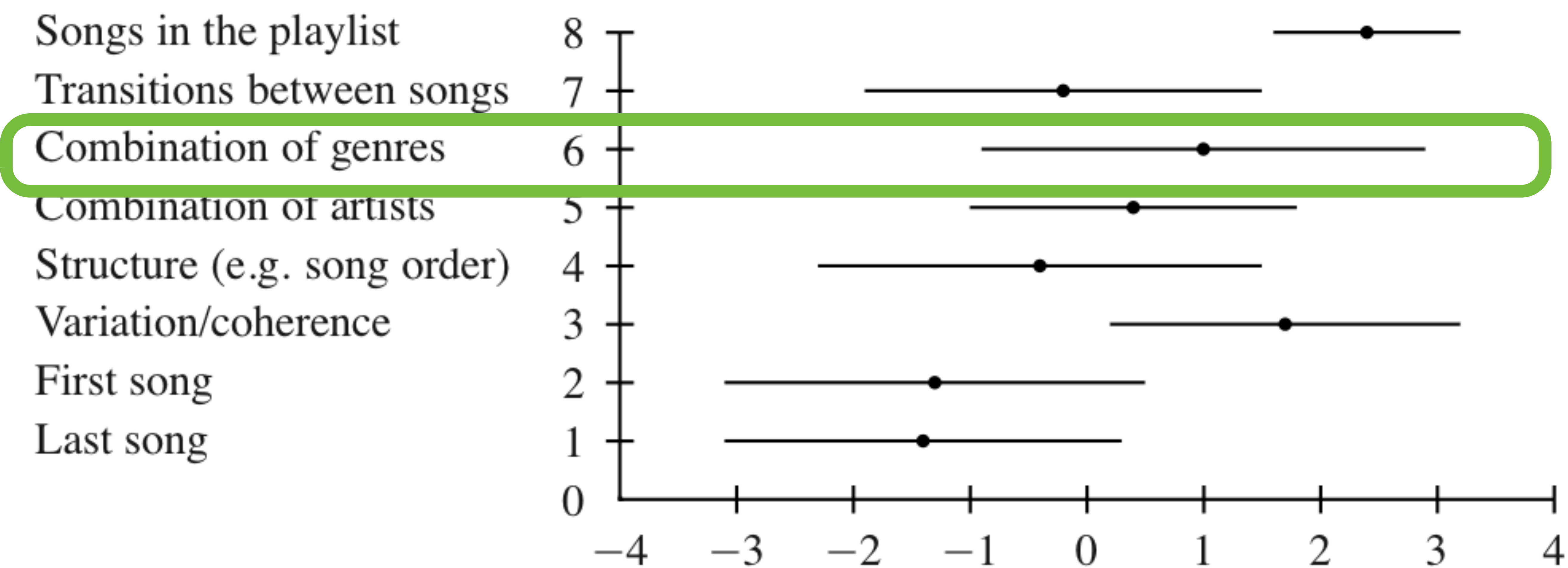


Figure 1: Importance of various factors in creating a playlist.

Survey with 14 participants

Learning Preferences for Music Playlists
A.M. de Mooij and W.F.J. Verhaegh

Factors affecting a good playlist

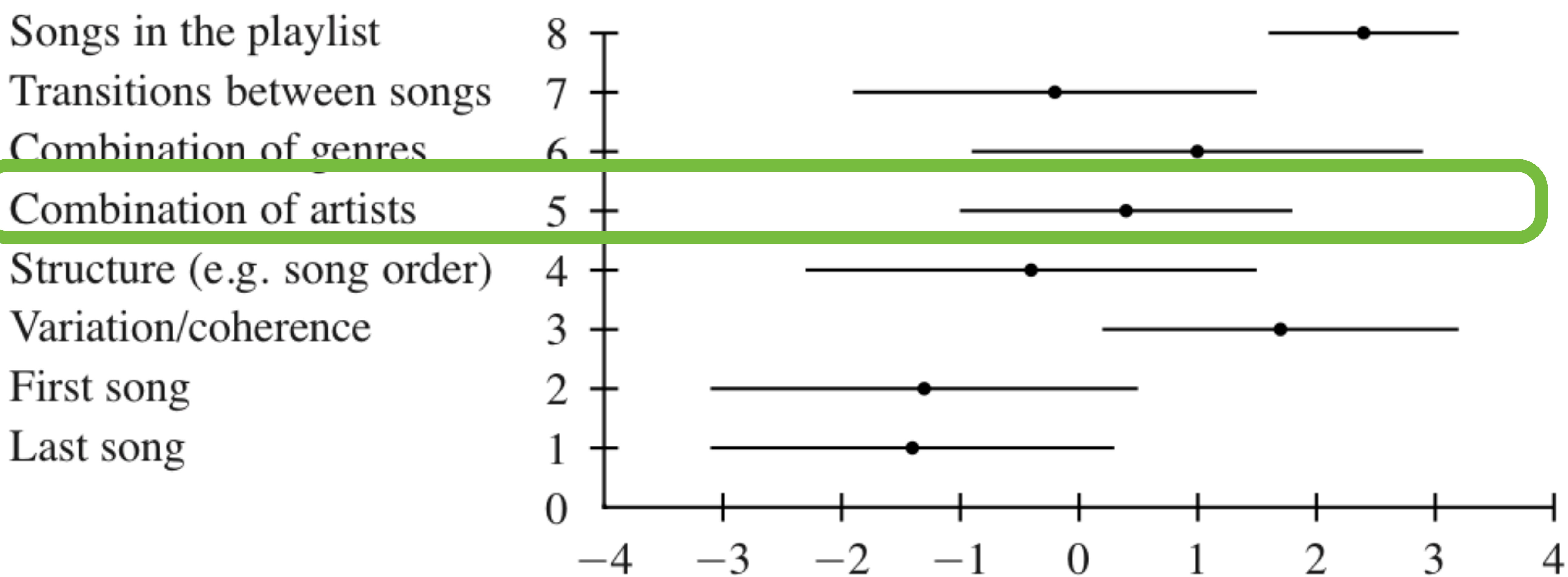


Figure 1: Importance of various factors in creating a playlist.

Survey with 14 participants

Factors affecting a good playlist

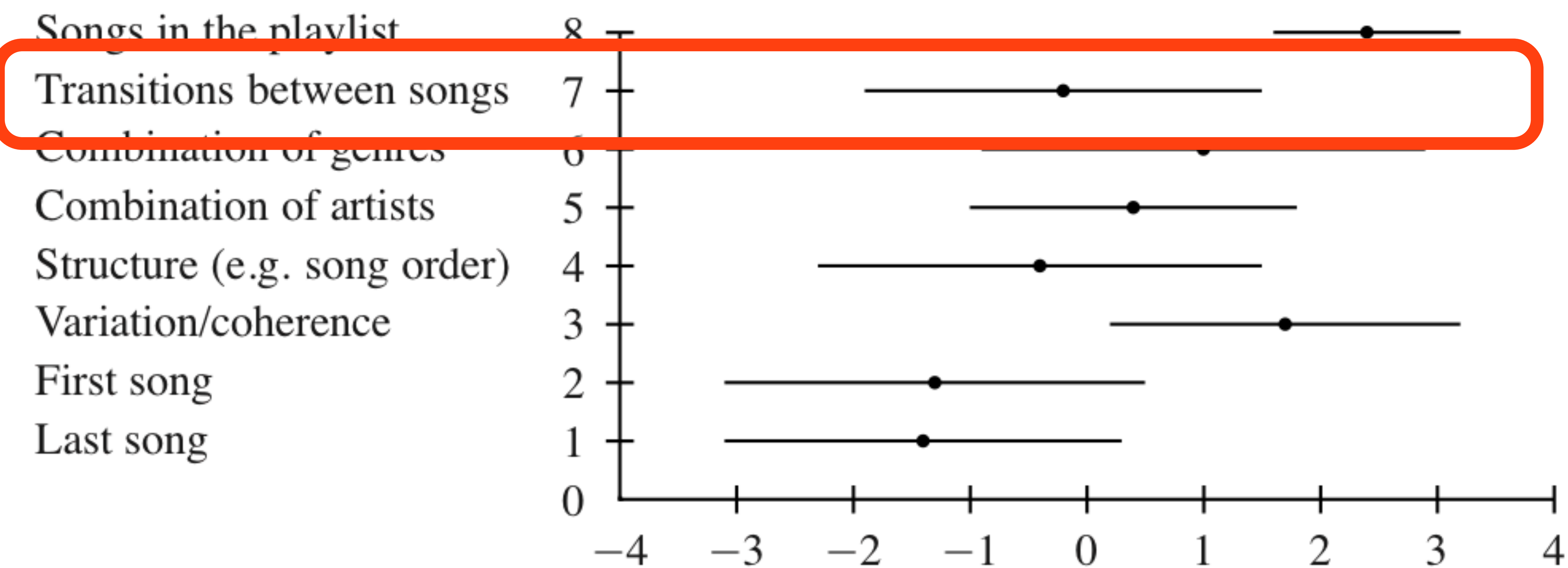


Figure 1: Importance of various factors in creating a playlist.

Survey with 14 participants

Factors affecting a good playlist

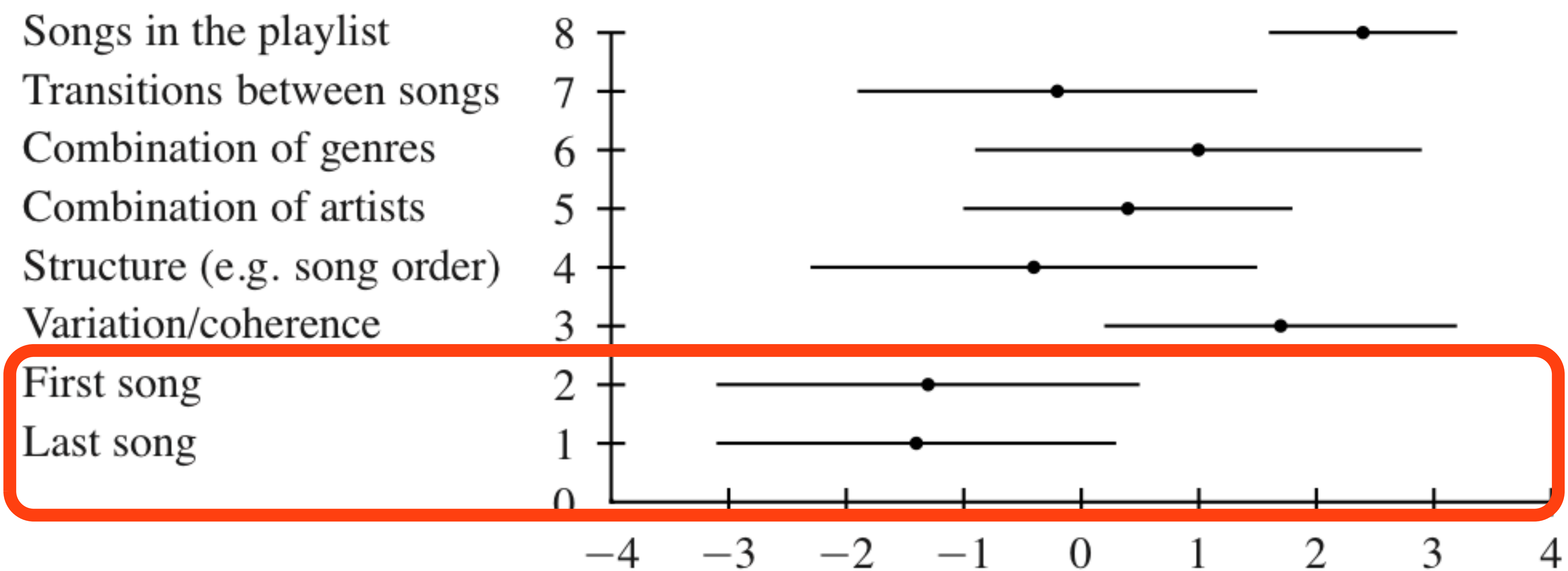


Figure 1: Importance of various factors in creating a playlist.

Survey with 14 participants

Factors affecting a good playlist

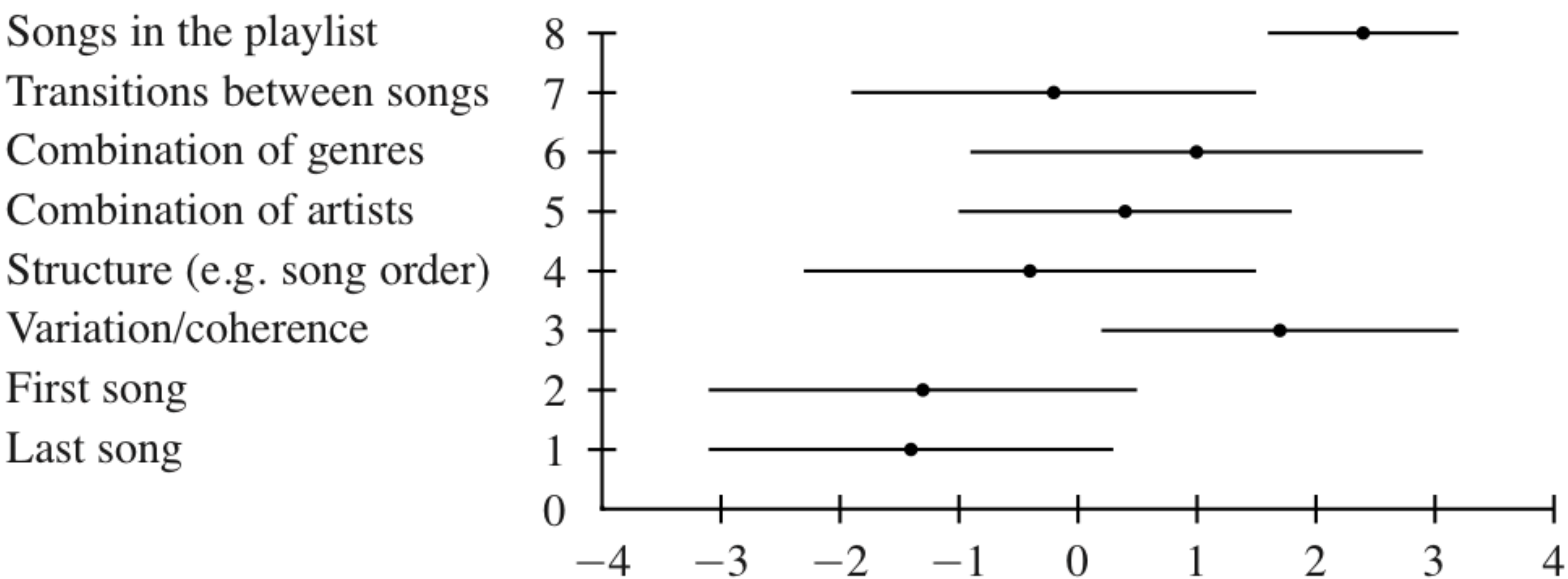


Figure 1: Importance of various factors in creating a playlist.

Survey with 14 participants

Factors affecting preference

- **Musical taste** - long term slowly evolving commitment to a genre
- Recent listening **history**
- **Mood** or state of mind
- The **context**:
  listening, driving, studying,
  working, exercising, etc.
- The **familiarity**
  - People sometimes prefer to listen to the familiar songs that they like less than non-familiar songs
  - Familiarity significantly predicts choice when controlling for the effects of liking, regret, and 'coolness'

I Want It Even Though I Do Not Like It: Preference for Familiar but Less Liked Music
Morgan K. Ward, Joseph K. Goodman, Julie R. Irwin

Coherence
Organizing principals for mix help requests

- Artist / Genre / Style
- Song similarity
- Event or activity
- Romance
- Message or story
- Mood
- Challenge or puzzle
- Orchestration
- Characteristic of the mix recipient
- Cultural references

Coherence
Organizing principals for mix help requests

- Artist / Genre / Style *"acoustic-country-folk type stuff",*
- Song similarity
- Event or activity
- Romance
- Message or story
- Mood
- Challenge or puzzle
- Orchestration
- Characteristic of the mix recipient
- Cultural references

Coherence
Organizing principals for mix help requests

- Artist / Genre / Style — *"acoustic-country-folk type stuff",*
- Song similarity
- Event or activity — *"anti-Valentine mix"*
- Romance
- Message or story
- Mood
- Challenge or puzzle
- Orchestration
- Characteristic of the mix recipient
- Cultural references

Coherence
Organizing principals for mix help requests

- Artist / Genre / Style
- Song similarity
- Event or activity
- Romance
- Message or story
- Mood
- Challenge or puzzle
- Orchestration
- Characteristic of the mix recipient
- Cultural references

"acoustic-country-folk type stuff",

"anti-Valentine mix"

a mix with the title "'quit being a douche', 'cause I'm in love with you."

Coherence
Organizing principals for mix help requests

- Artist / Genre / Style
- Song similarity
- Event or activity
- Romance
- Message or story
- Mood
- Challenge or puzzle
- Orchestration
- Characteristic of the mix recipient
- Cultural references

“acoustic-country-folk type stuff”,

“anti-Valentine mix”

a mix with the title “‘quit being a douche’, ’cause I’m in love with you.”

song whose title is a question?

Coherence
Organizing principals for mix help requests

- Artist / Genre / Style — *"acoustic-country-folk type stuff",*
- Song similarity
- Event or activity — *"anti-Valentine mix"*
- Romance
- Message or story — *a mix with the title "'quit being a douche', 'cause I'm in love with you."*
- Mood
- Challenge or puzzle — *song whose title is a question?*
- Orchestration — *songs where the singer hums for a little bit*
- Characteristic of the mix recipient
- Cultural references

Coherence
Organizing principals for mix help requests

- Artist / Genre / Style — *“acoustic-country-folk type stuff”,*
- Song similarity
- Event or activity — *“anti-Valentine mix”*
- Romance
- Message or story — *a mix with the title “‘quit being a douche’, ’cause I’m in love with you.”*
- Mood
- Challenge or puzzle — *song whose title is a question?*
- Orchestration — *songs where the singer hums for a little bit*
- Characteristic of the mix recipient
- Cultural references — *“songs about superheroes”*

“People have gotten used to listening to songs in the order they want, and they'll want to continue to do so even if they can't get the individual songs from file-trading programs.”

Phil Leigh

Ordering Principals

- Bucket of similars, genre
- Acoustic attributes such as tempo, loudness, danceability
- Social attributes such popularity, 'hotness'
- Mood attributes ('sad' to 'happy')
- Theme / lyrics
- Alphabetical
- Chronological
- Random
- Song transitions
- Novelty orderings

1	All That I'm Living For
2	Away
3	Away
4	Away (live)
5	Believe
6	Break My Fall
7	Breakdown
8	Breath
9	Breathe No More [Live]
10	Bring Me To Life
11	Bring Me To Life (Live)
12	Bring Me To Life [Live]
13	Call Me When You're Sober
14	Call Me When You're Sober
15	Cloud Nine
16	Cloud Nine

Novelty ordering

```
 0    We Wish You A Merry Christmas - Weezer
 1    Stranger Things Have Happened - Foo Fighters
 2    Dude We're Finally Landing - Rivers Cuomo
 3    Gotta Be Somebody's Blues - Jimmy Eat World
 4    Someday You Will Be Loved - Death Cab For Cutie
 5    Dancing In The Moonlight - The Smashing Pumpkins
 6    Take The Long Way Round - Teenage Fanclub
 7    Don't Make Me Prove It - Veruca Salt
 8    The Sacred And Profane - Smashing Pumpkins, The
 9    Everything Is Alright - Motion City Soundtrack
10    Trains, brains & rain - The Flaming Lips
11    No One Needs To Know - Ozma
12    What Is Your Secret - Nada Surf
13    The Spark That Bled - Flaming Lips, The
14    Defending The Faith - Nerf Herder
```

Novelty ordering

```
 0    We Wish You A Merry Christmas - Weezer
 1    Stranger Things Have Happened - Foo Fighters
 2    Dude We're Finally Landing - Rivers Cuomo
 3    Gotta Be Somebody's Blues - Jimmy Eat World
 4    Someday You Will Be Loved - Death Cab For Cutie
 5    Dancing In The Moonlight - The Smashing Pumpkins
 6    Take The Long Way Round - Teenage Fanclub
 7    Don't Make Me Prove It - Veruca Salt
 8    The Sacred And Profane - Smashing Pumpkins, The
 9    Everything Is Alright - Motion City Soundtrack
10    Trains, brains & rain - The Flaming Lips
11    No One Needs To Know - Ozma
12    What Is Your Secret - Nada Surf
13    The Spark That Bled - Flaming Lips, The
14    Defending The Faith - Nerf Herder
```

Novelty ordering

```
 0    We Wish You A Merry Christmas - Weezer
 1    Stranger Things Have Happened - Foo Fighters
 2    Dude We're Finally Landing - Rivers Cuomo
 3    Gotta Be Somebody's Blues - Jimmy Eat World
 4    Someday You Will Be Loved - Death Cab For Cutie
 5    Dancing In The Moonlight - The Smashing Pumpkins
 6    Take The Long Way Round - Teenage Fanclub
 7    Don't Make Me Prove It - Veruca Salt
 8    The Sacred And Profane - Smashing Pumpkins, The
 9    Everything Is Alright - Motion City Soundtrack
10    Trains, brains & rain - The Flaming Lips
11    No One Needs To Know - Ozma
12    What Is Your Secret - Nada Surf
13    The Spark That Bled - Flaming Lips, The
14    Defending The Faith - Nerf Herder
```

Novelty ordering

```
 0    We Wish You A Merry Christmas - Weezer
 1    Stranger Things Have Happened - Foo Fighters
 2    Dude We're Finally Landing - Rivers Cuomo
 3    Gotta Be Somebody's Blues - Jimmy Eat World
 4    Someday You Will Be Loved - Death Cab For Cutie
 5    Dancing In The Moonlight - The Smashing Pumpkins
 6    Take The Long Way Round - Teenage Fanclub
 7    Don't Make Me Prove It - Veruca Salt
 8    The Sacred And Profane - Smashing Pumpkins, The
 9    Everything Is Alright - Motion City Soundtrack
10    Trains, brains & rain - The Flaming Lips
11    No One Needs To Know - Ozma
12    What Is Your Secret - Nada Surf
13    The Spark That Bled - Flaming Lips, The
14    Defending The Faith - Nerf Herder
```

Novelty ordering

```
 0    We Wish You A Merry Christmas - Weezer
 1    Stranger Things Have Happened - Foo Fighters
 2    Dude We're Finally Landing - Rivers Cuomo
 3    Gotta Be Somebody's Blues - Jimmy Eat World
 4    Someday You Will Be Loved - Death Cab For Cutie
 5    Dancing In The Moonlight - The Smashing Pumpkins
 6    Take The Long Way Round - Teenage Fanclub
 7    Don't Make Me Prove It - Veruca Salt
 8    The Sacred And Profane - Smashing Pumpkins, The
 9    Everything Is Alright - Motion City Soundtrack
10    Trains, brains & rain - The Flaming Lips
11    No One Needs To Know - Ozma
12    What Is Your Secret - Nada Surf
13    The Spark That Bled - Flaming Lips, The
14    Defending The Faith - Nerf Herder
```

Where song order rules
The Dance DJ

- For the Dance DJ - song order and transitions are especially important
- Primary goal: make people dance
- How?
  - Selecting
    - tracks that mix well
    - takes the audience on a journey
    - audience feedback is important
  - Mixing
    - seamless song transitions

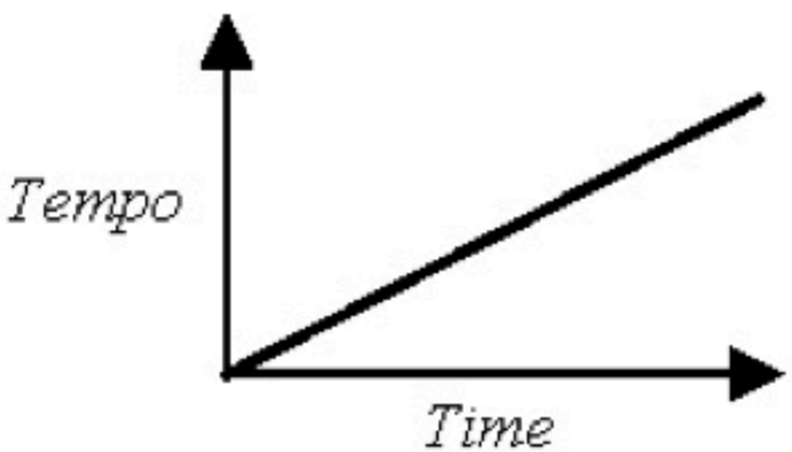
Is the DJ an Artist?

Is a mixset a piece of art?

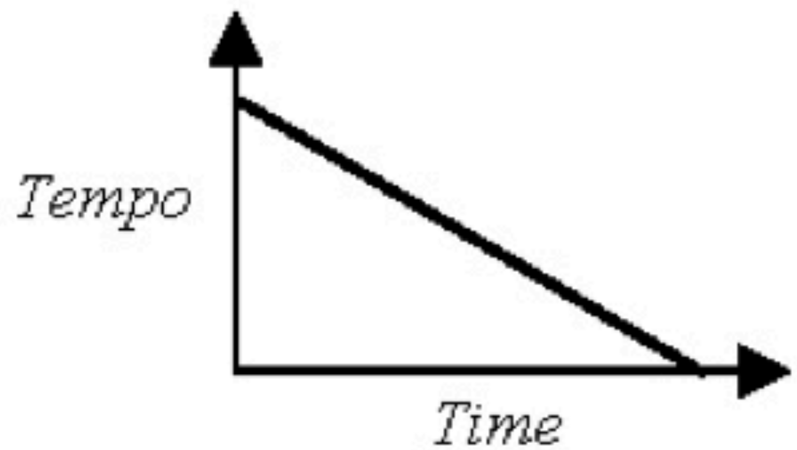
Hang the DJ: Automatic Sequencing and Seamless Mixing of Dance-Music Tracks
Dave Cliff Publishing Systems and Systems Laboratory HP Laboratories Bristol HPL-2000-104 9th August, 2000*

By BRENT SILBY

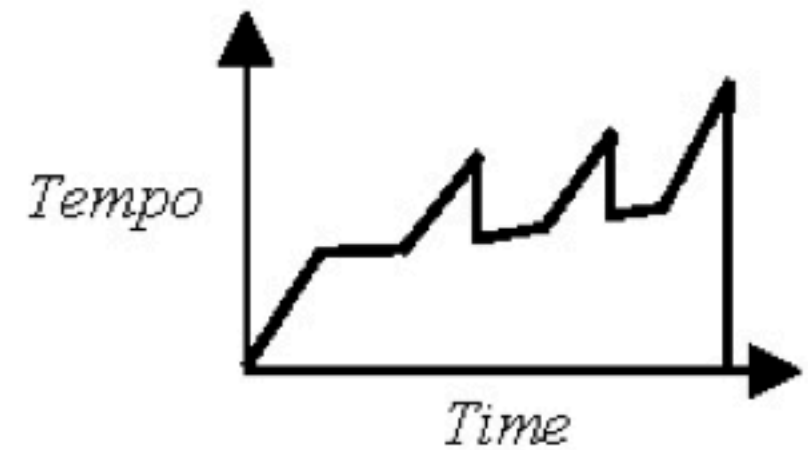
Tempo Trajectories



Warmup

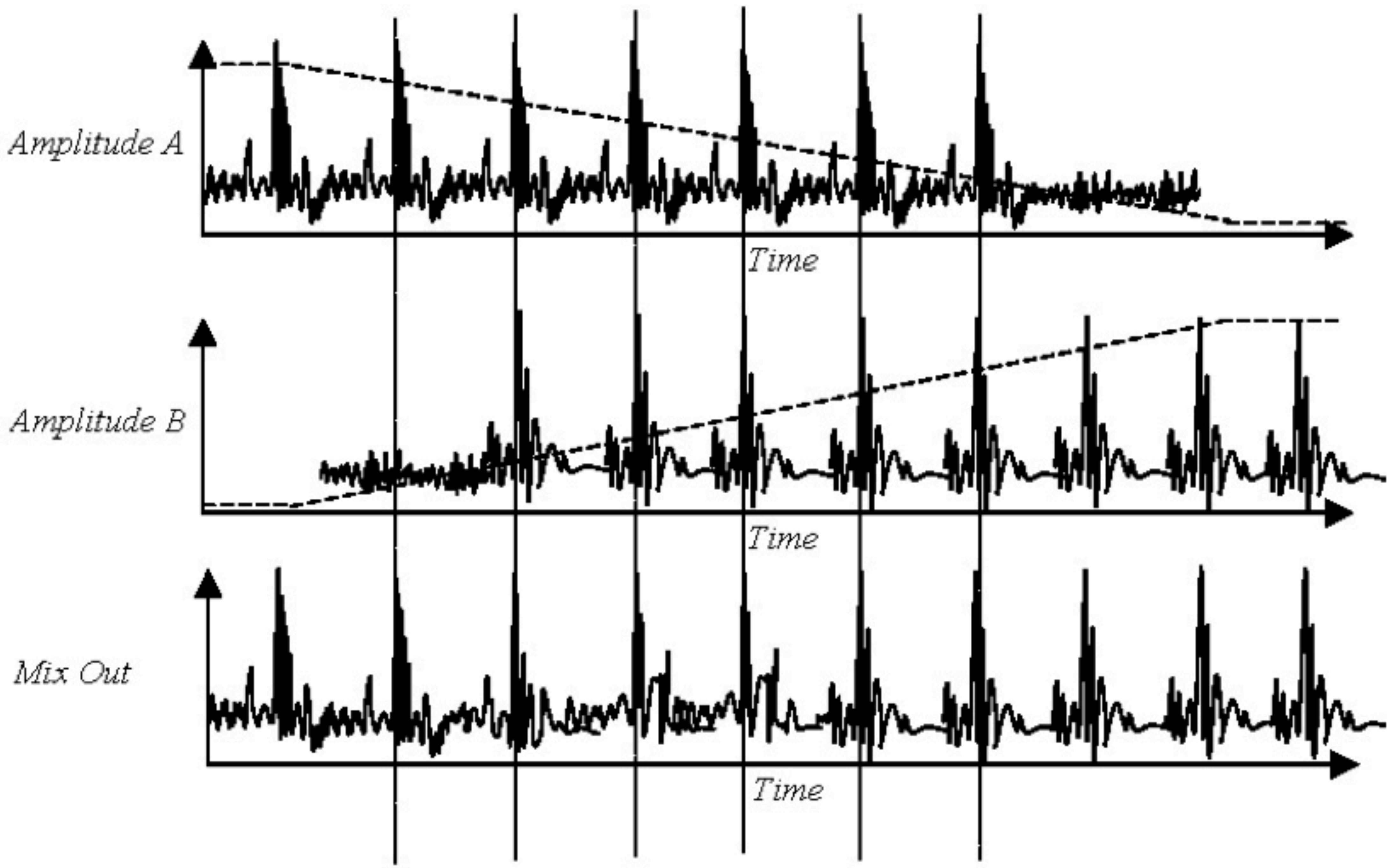


Cool down



Nightclub

Coherence
Song to Song



Beat Matching and Cross-fading

Don’t underestimate the power of the shuffle

Don’t underestimate the power of the shuffle

laugh-out-loud pleasurable

Don’t underestimate the power of the shuffle

white-knuckle ride

Don’t underestimate the power of the shuffle

“...teaches me connections between disparate kinds of music and the infinite void. I understand the universe better”

Don’t underestimate the power of the shuffle

...forge(ing) new connections between my heart and my ears

Don’t underestimate the power of the shuffle

each randomly-sequenced track like an aural postcard

Don’t underestimate the power of the shuffle

had made me re-examine things I thought I knew about my favourite music

Don’t underestimate the power of the shuffle

...hear(ing) songs that I haven’ t heard for years and fall (ing) in love with them again

Don’t underestimate the power of the shuffle

Random shuffle can turn large music libraries into an ‘Aladdin’ s cave’ of aural surprises

Don’t underestimate the power of the shuffle

...the random effect delivers a sequence of music so perfectly thematically 'in tune' that (it) is quite unsettling

Serendipity of the shuffle

Finding meaningful experience in chance encounters

- Serendipity can improve the listening experience
- Choosing songs randomly from a personal collection can yield serendipitous listening
- Drawing from too large, or too small of a collection reduces serendipity

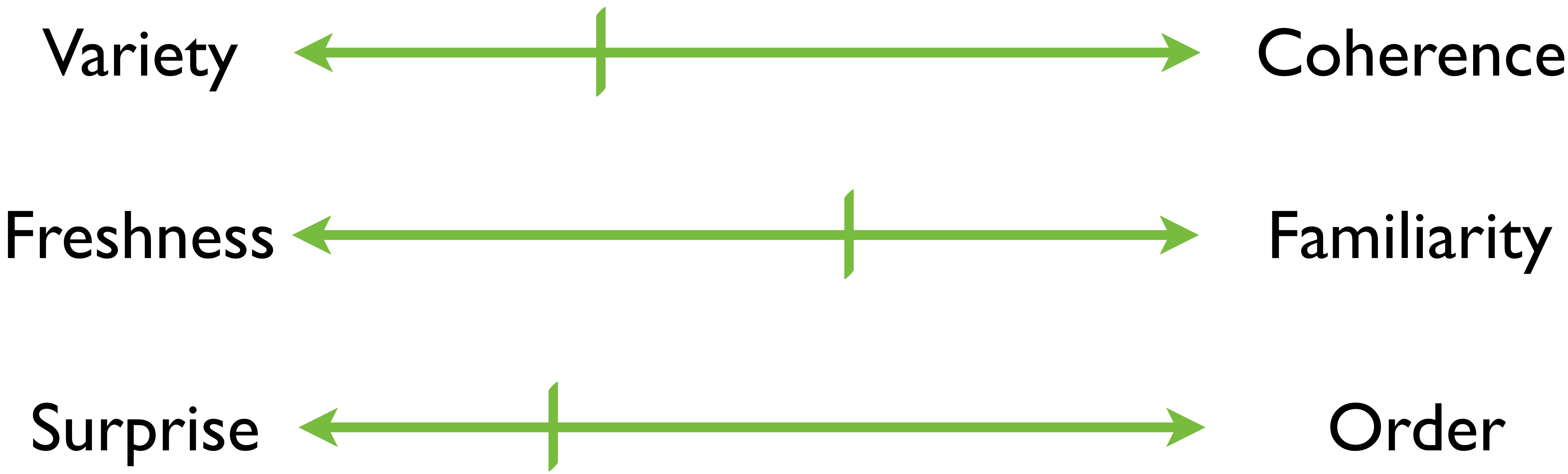
People like shuffle play

		content organisation		
		constrained	unconstrained	
preferred listening	shuffle	22	69	91
	both	4	4	8
	sequential	13	1	14
		39	74	113

Table 1. Preferred listening mode (shuffle or sequential) and organisation of music content (constrained or unconstrained)

People shuffle genres, albums and playlists

Playlist tradeoffs



Different listeners have different optimal settings
Mood and context can affect optimal settings

Playlist Variety
A good playlist is not a bag of similar tracks

Worklist | Playlist | Recent

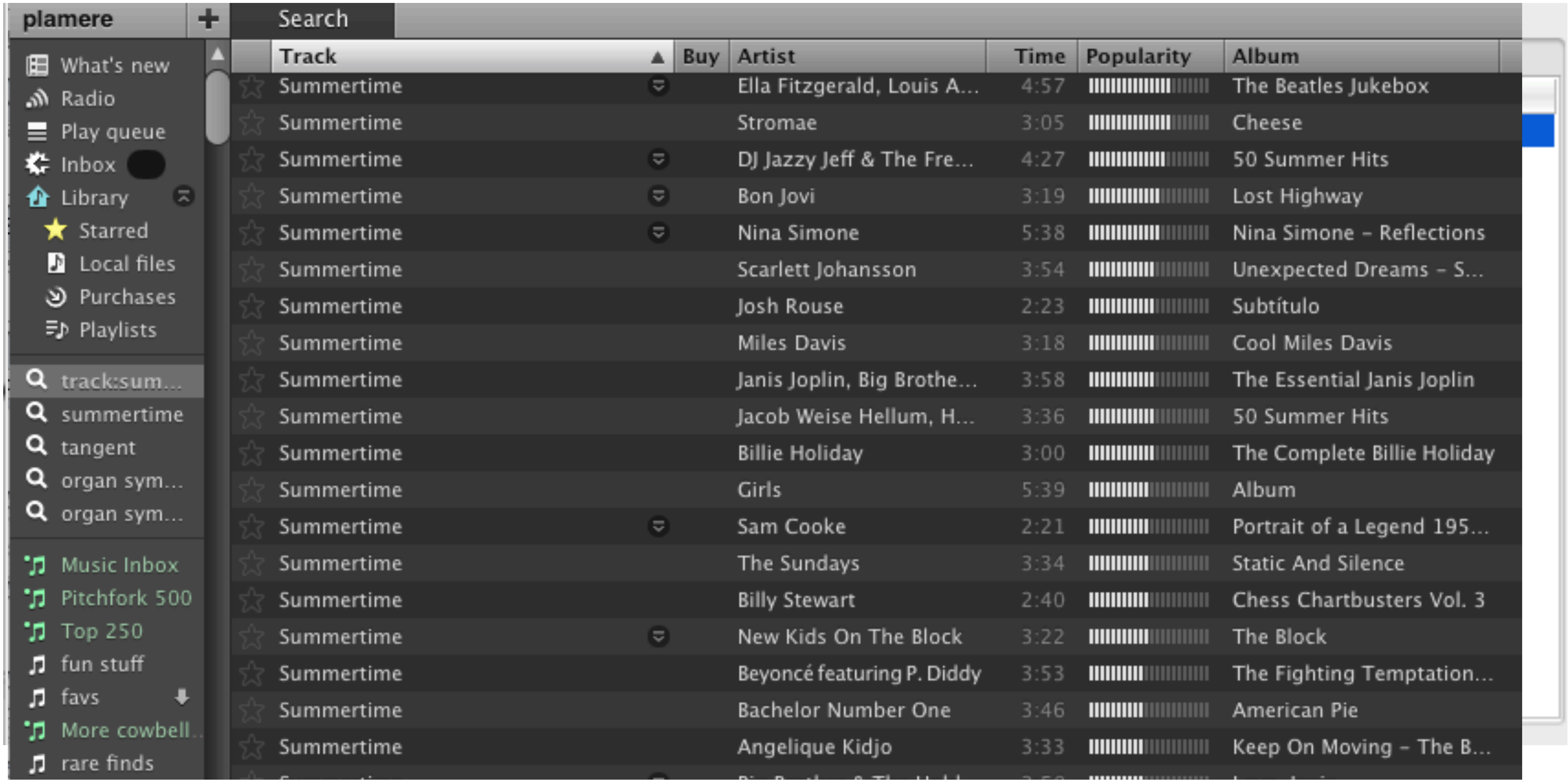
#	Track	Album	Artist	Genre
1	Farrakorn	Party Patrol	Pizzle	Punk
2	What's Wrong with my foot?	Party Patrol	Pizzle	Punk
3	I love her to Pieces	Party Patrol	Pizzle	Punk
4	In my livid eyes	Party Patrol	Pizzle	Punk
5	A little exposure	Party Patrol	Pizzle	Punk
6	Donkey Punch	Party Patrol	Pizzle	Punk
7	Wow!	Gimme Some	Nova Express	Punk
8	Flowers on the Wall	Party Patrol	Pizzle	Punk
9	Wet Brain	Party Patrol	Pizzle	Punk
10	Tammy ate a bad piece of pork	Party Patrol	Pizzle	Punk
11	Pucker String	Party Patrol	Pizzle	Punk
12	Pizzle: Party Patrol	High Energy Rock and Roll	Magnatune Compilation	Rock
13	Nunchukkaboot	Party Patrol	Pizzle	Punk
14	Party Patrol	Party Patrol	Pizzle	Punk
15	Motorway	Gimme Some	Nova Express	Punk

Playlist Variety
A good playlist is not a bag of similar tracks

Worklist | Playlist | Recent

#	Track	Album	Artist	Genre
1	Pizzle: In my livid eyes	High Energy Rock and Roll	Magnatune Compilation	Rock
2	In my livid eyes	Party Patrol	Pizzle	Punk
3	Wow!	Gimme Some	Nova Express	Punk
4	Euthanize Tunnel Zone	Hellavator Musick	Skitzo	Metal
5	Hostage Situation	Listen Up, Baby!	Electric Frankenstein	Punk
6	Dirty brown duster	Jacksploitation	Jackalopes	Punk
7	Park that ass	Geeking Dream	The Strap Ons	Punk
8	Higher education	Thrill Hype	The Napoleon Blown Aparts	Punk Rock
9	KC rip off	Up from the mud	Spinecar	Hard Rock
10	As it Descends	Night of the Black Wyvern	Utopia Banished	Metal
11	No Cure	8 Seconds	Pain Factor	Metal
12	Everyday Like Saturday (bonu...	Middle Age Suicide	Rocket City Riot	Rock
13	Function	Trancelucent	Somadrone	Rock
14	Feverdream #1	Alpha & Oranges	Atomic Opera	Hard Rock
15	Look And Feel Years Younger	I Don't Know What I'm Doing	Brad Sucks	Rock

Playlist Variety
A good playlist is not a bag of similar tracks



Playlisting is not Recommendation

Playlisting is not Recommendation

Recommendation	Playlist

Playlisting is not Recommendation

Recommendation	**Playlist**
Primarily for music discovery	Primarily for music listening

Playlisting is not Recommendation

Recommendation	Playlist
Primarily for music discovery	Primarily for music listening
Minimize familiar artists	Familiar artists in abundance

Playlisting is not Recommendation

Recommendation	**Playlist**
Primarily for music discovery	Primarily for music listening
Minimize familiar artists	Familiar artists in abundance
Order not important	Order can be critical

Playlisting is not Recommendation

Recommendation	Playlist
Primarily for music discovery	Primarily for music listening
Minimize familiar artists	Familiar artists in abundance
Order not important	Order can be critical
Limited Context (shopping)	Rich contexts - party, jogging, working, gifts

Playlisting is not Recommendation

Recommendation	**Playlist**
Primarily for music discovery	Primarily for music listening
Minimize familiar artists	Familiar artists in abundance
Order not important	Order can be critical
Limited Context (shopping)	Rich contexts - party, jogging, working, gifts

However, playlists may be better vector for music discovery than traditional recommendation

Playlisting nuts and bolts
formats and rules

Playlist formats

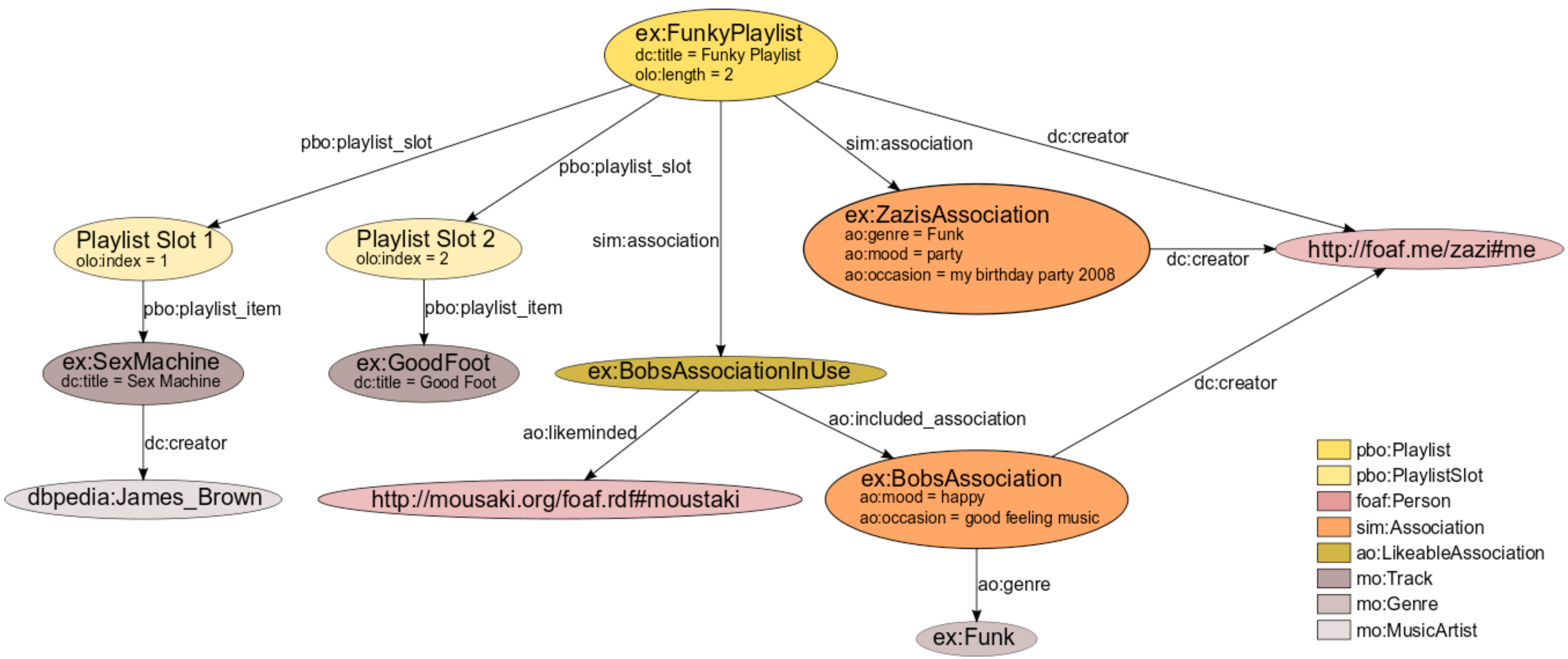
- Lots of formats - Some notable examples:
  - M3U - simple list of files - one per line
  - XSPF - ‘spiff’ - XML based format
  - The Playback Ontology
- Resources:
  - http://microformats.org/wiki/audio-info-formats
  - http://lizzy.sourceforge.net/docs/formats.html
  - http://gonze.com/playlists/playlist-format-survey.html

Example XSPF

```xml
<?xml version="1.0" encoding="UTF-8"?>
<playlist version="1" xmlns="http://xspf.org/ns/0/">
    <trackList>
        <track>
            <location>http://example.com/song_1.mp3</location>
            <creator>Led Zeppelin</creator>
            <album>Houses of the Holy</album>
            <title>No Quarter</title>
            <annotation>I love this song</annotation>
            <duration>271066</duration>
            <image>http://images.amazon.com/images/P/B000002J0B.jpg</image>
            <info>http://example.com</info>
        </track>
       <track>
            <location>http://example.com/song_1.mp3</location>
            <creator>Led Zeppelin</creator>
            <album>ii</album>
            <title>No Quarter</title>
            <annotation>This one too</annotation>
            <duration>271066</duration>
            <image>http://images.amazon.com/images/P/B000002J0B.jpg</image>
            <info>http://example.com</info>
        </track>
    </trackList>
</playlist>
```

The Playback Ontology

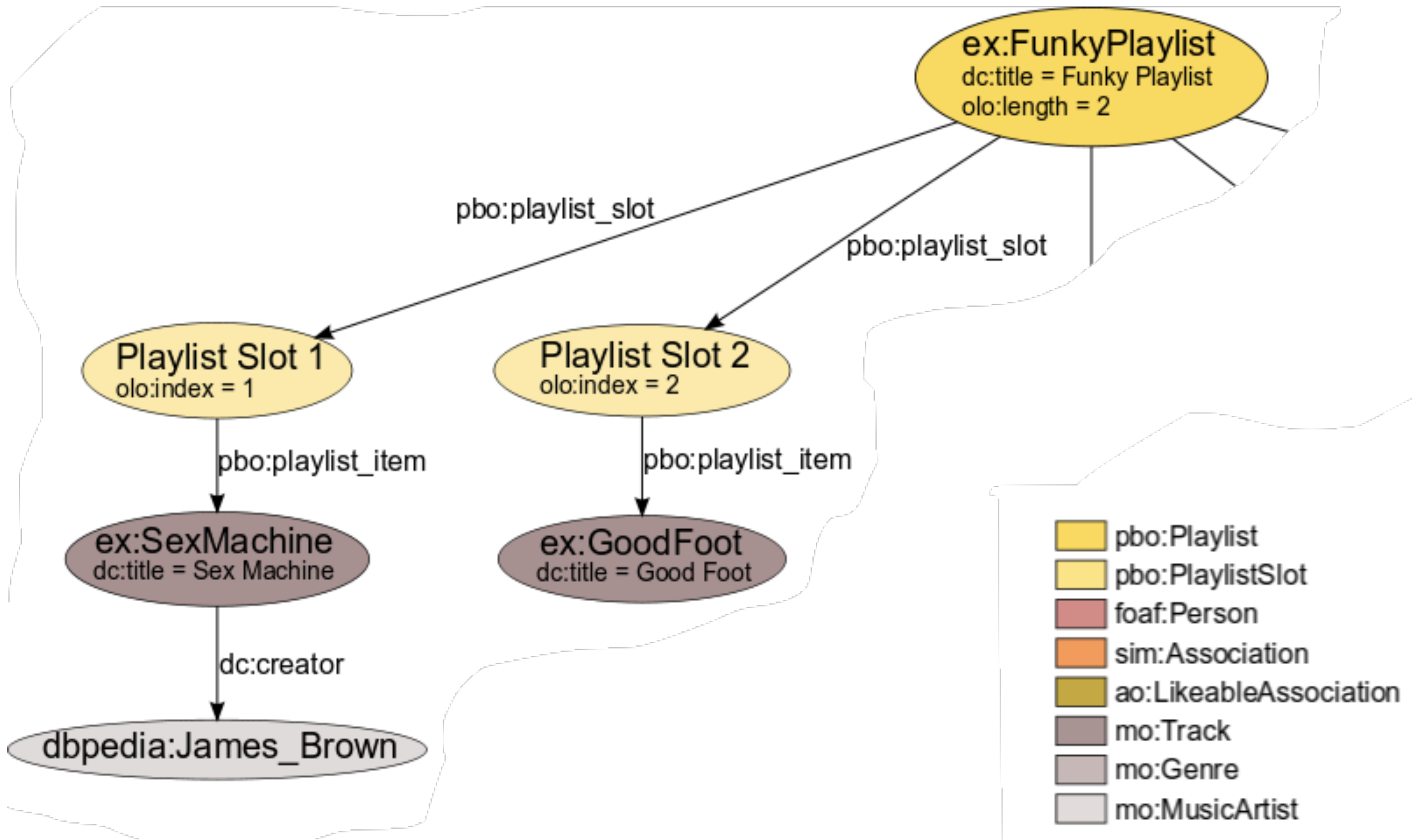
The *Play Back Ontology* provides basic concepts and properties for describing concepts that are related to the *play back domain*, e.g. a playlist,play back and skip counter, on/ for the Semantic Web.



http://smiy.sourceforge.net/pbo/spec/playbackontology.html http://smiy.wordpress.com/2010/07/27/the-play-back-ontology/

The Playback Ontology

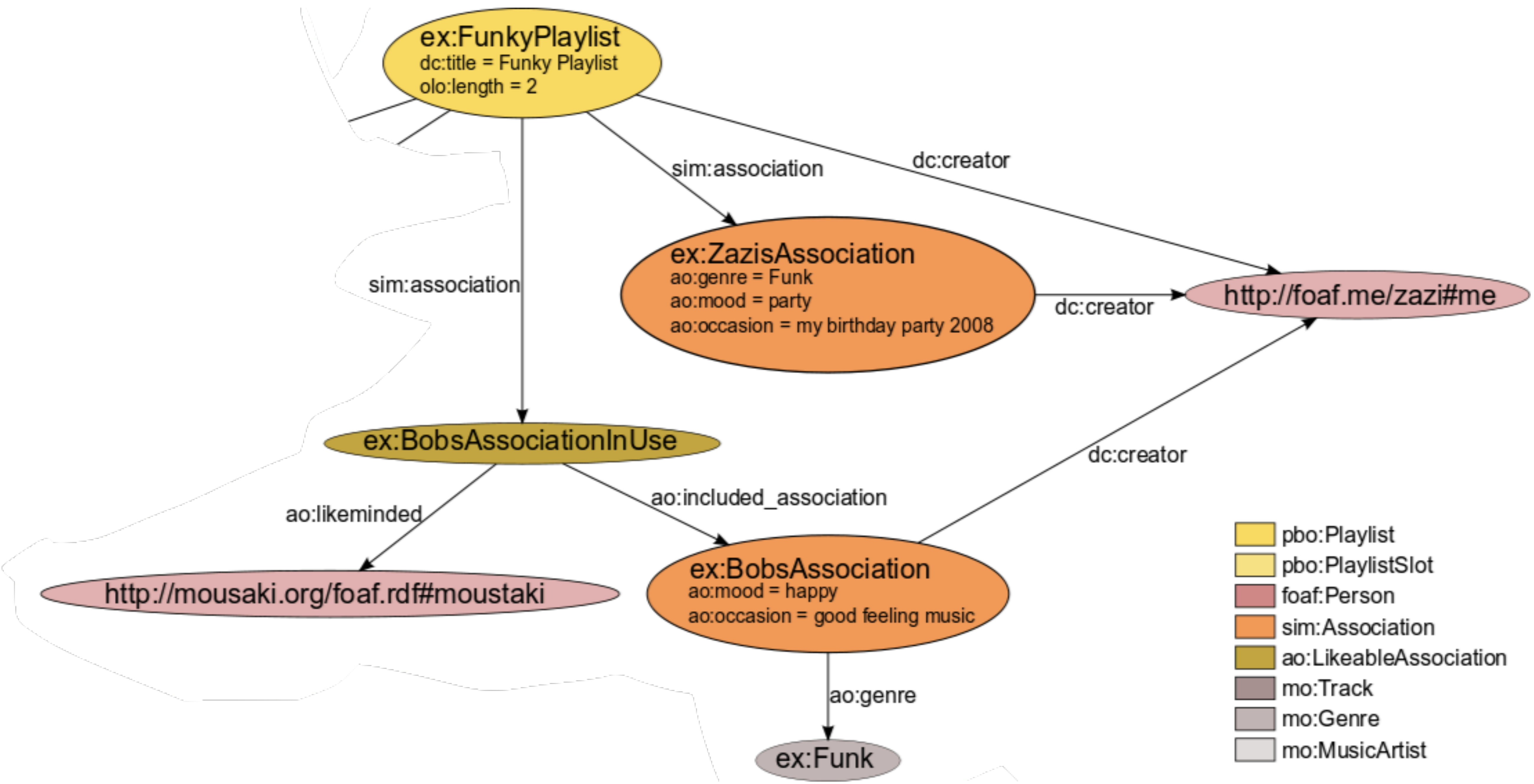
Modeling items in the playlist by extending the ordered list ontology



http://smiy.sourceforge.net/pbo/spec/playbackontology.html http://smiy.wordpress.com/2010/07/27/the-play-back-ontology/

The Playback Ontology

Expressing similarity and creation provenance



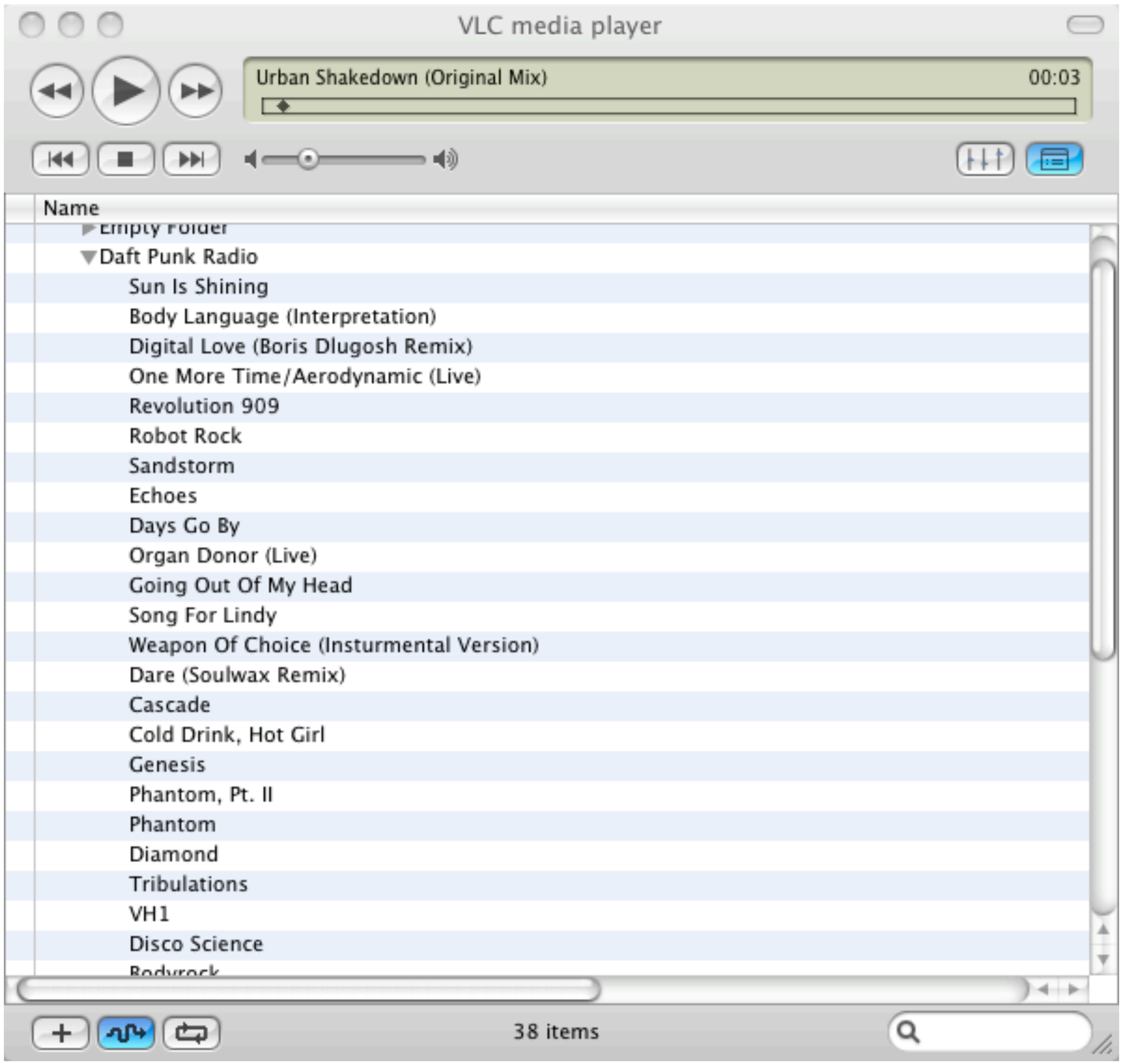
http://smiy.sourceforge.net/pbo/spec/playbackontology.html http://smiy.wordpress.com/2010/07/27/the-play-back-ontology/

Survey of playlisting systems and tools


Social
RADIO PARADISE
Listener Supported
iLike
playlist.com
partyStrands
Spotify
last.fm
mixpod
Manual
Automated
PANDORA
Non-Social

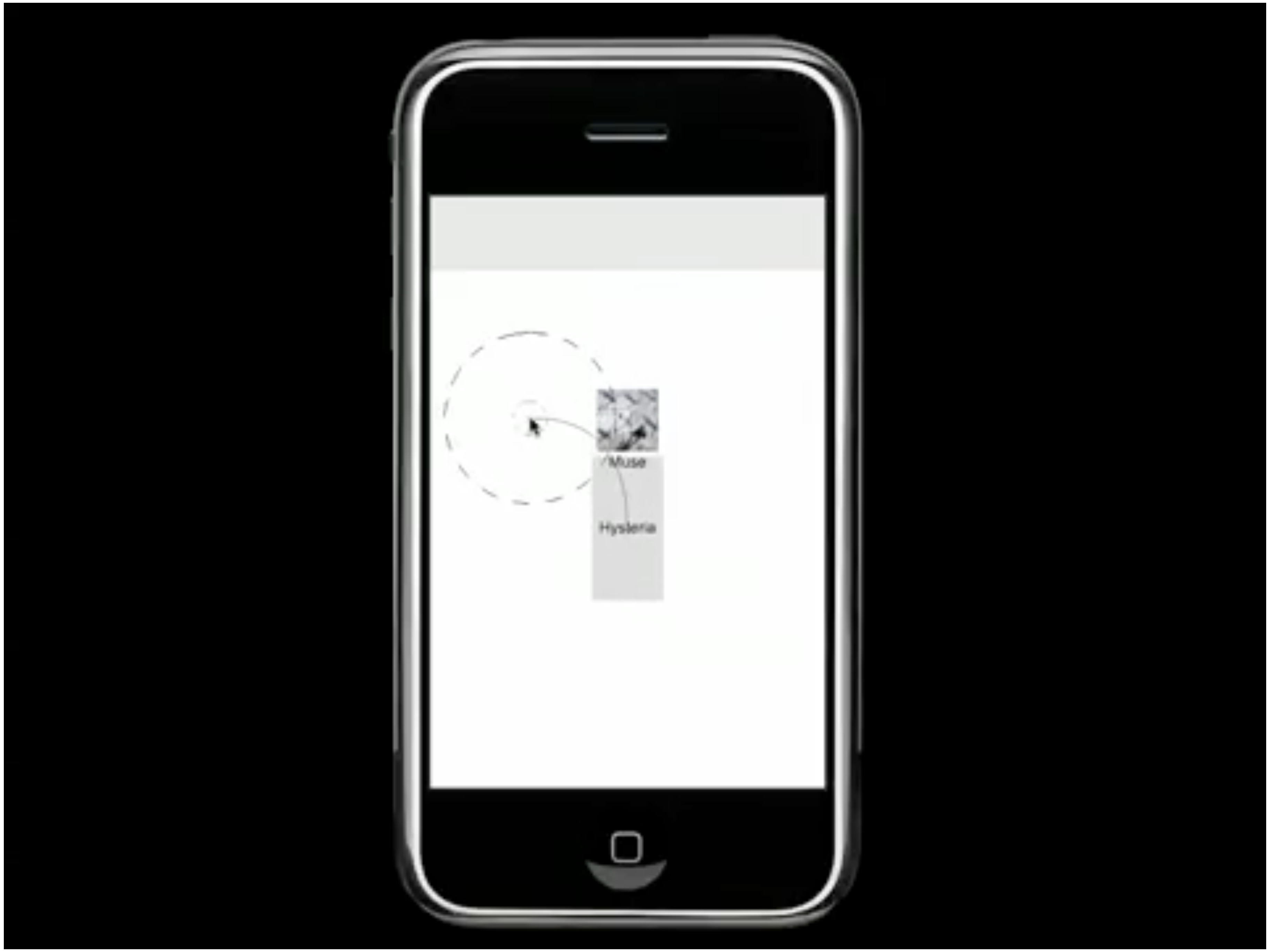

Social
Manual
Automated
Non-Social
RADIO PARADISE
playlist.com
Spotify
partyStrands
last.fm
iLike
mixpod
PANDORA

Manual Non-Social

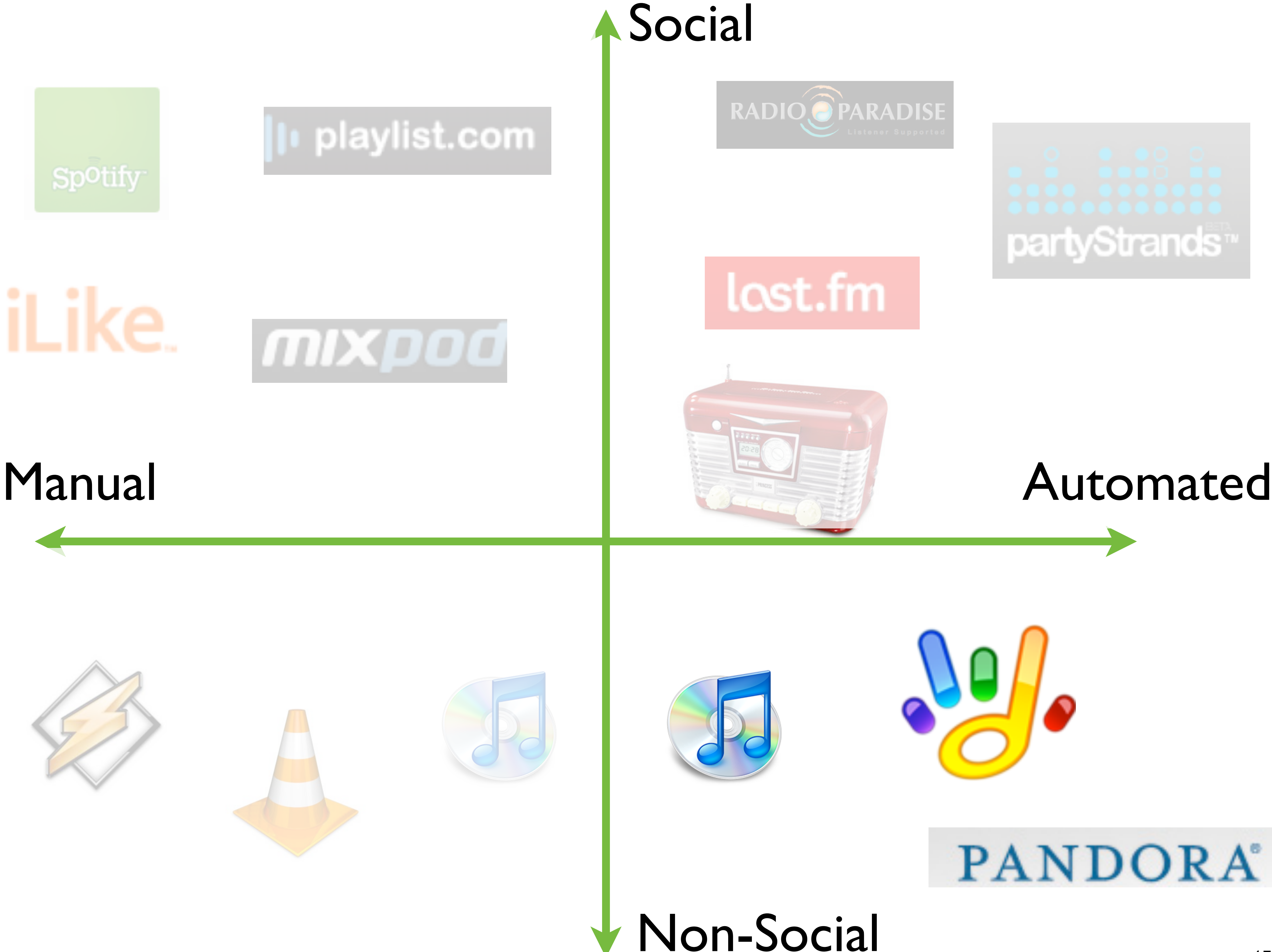


Rush: Repeated Recommendations on Mobile Devices

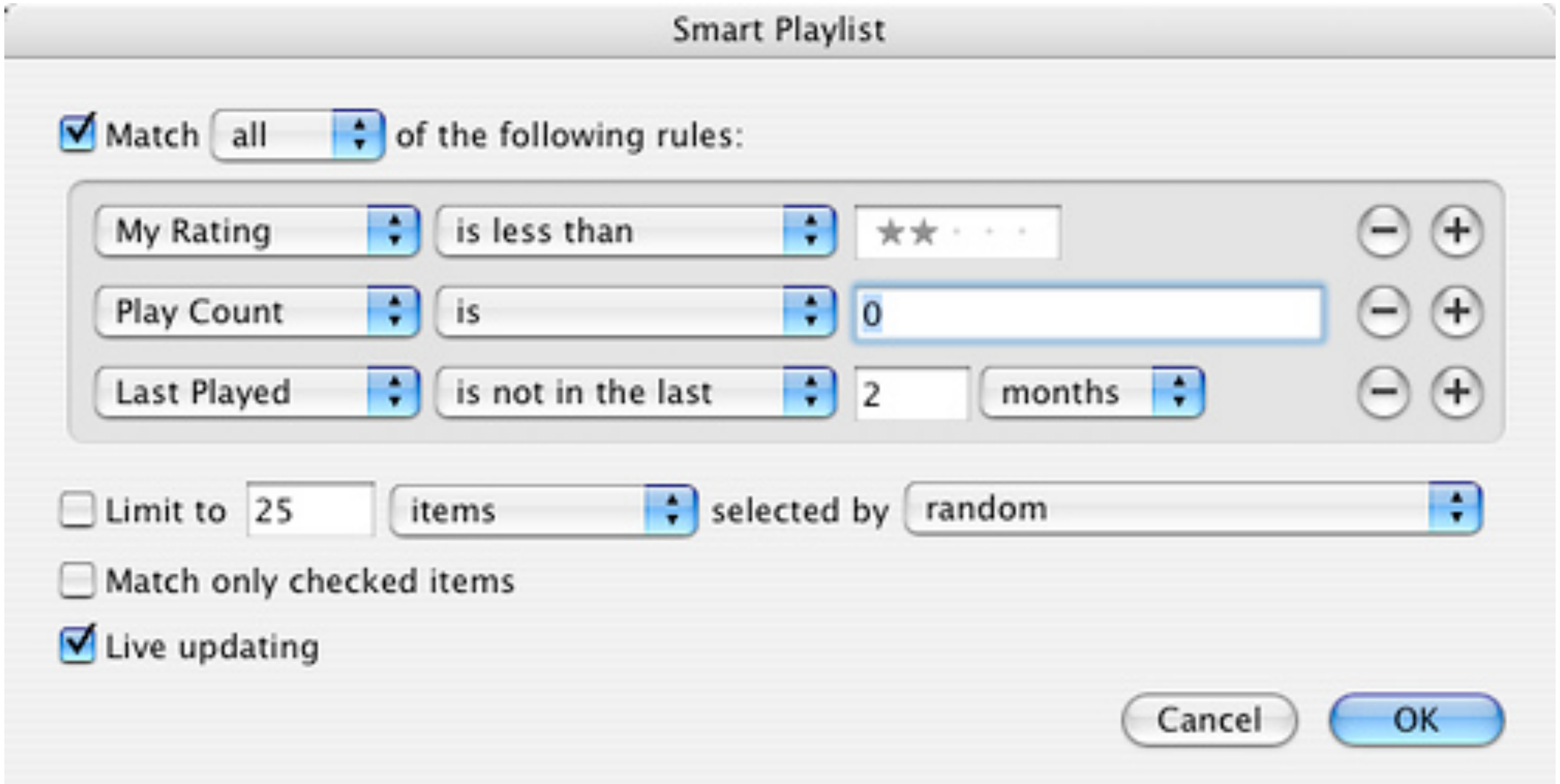
Rush: Repeated Recommendations on Mobile Devices



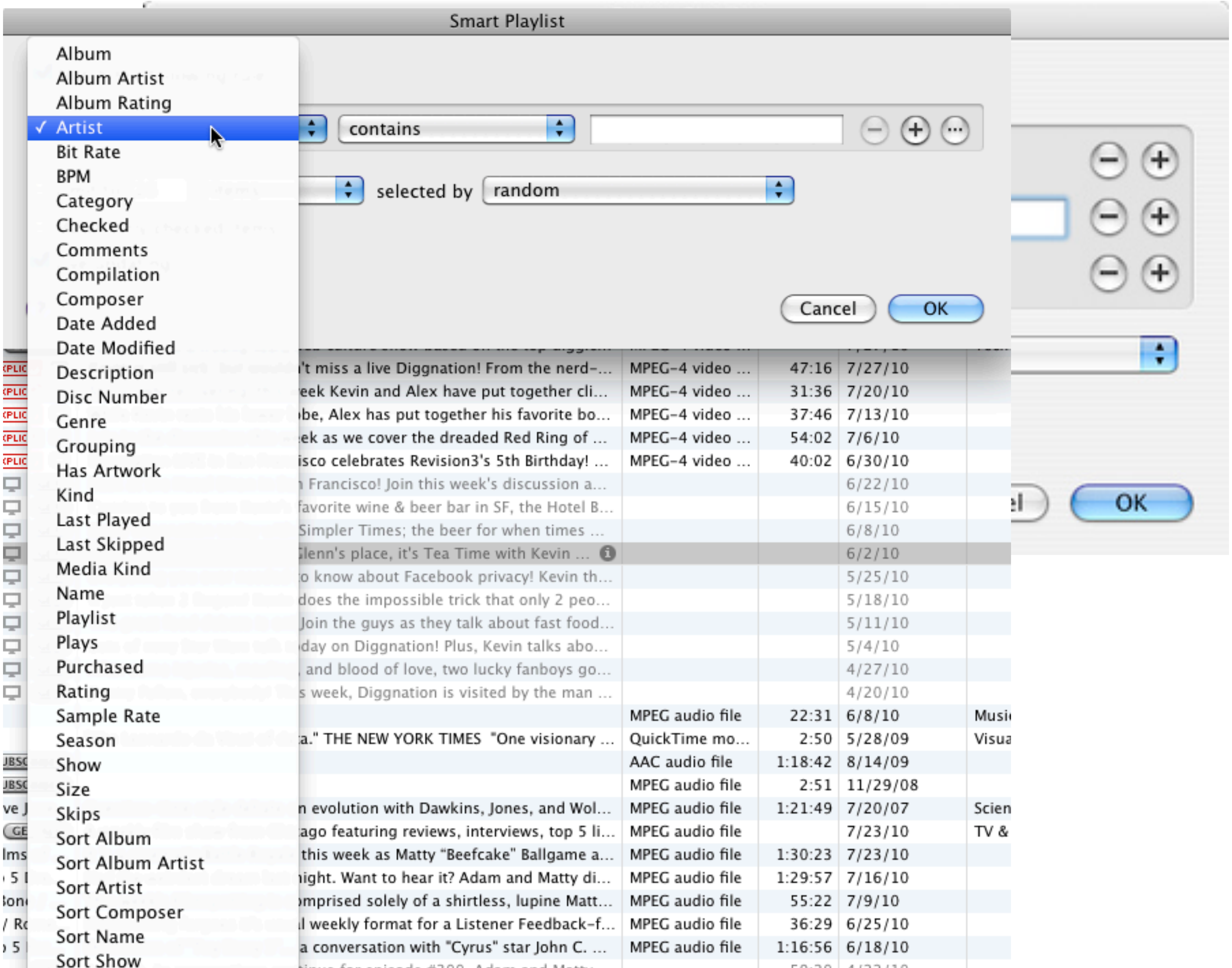
Rush: Repeated Recommendations on Mobile Devices
Dominikus Baur, Sebastian Boring, Andreas Butz


Social
Spotify
playlist.com
RADIO PARADISE
Listener Supported
partyStrands
iLike
mixpod
last.fm
Manual
Automated
PANDORA
Non-Social

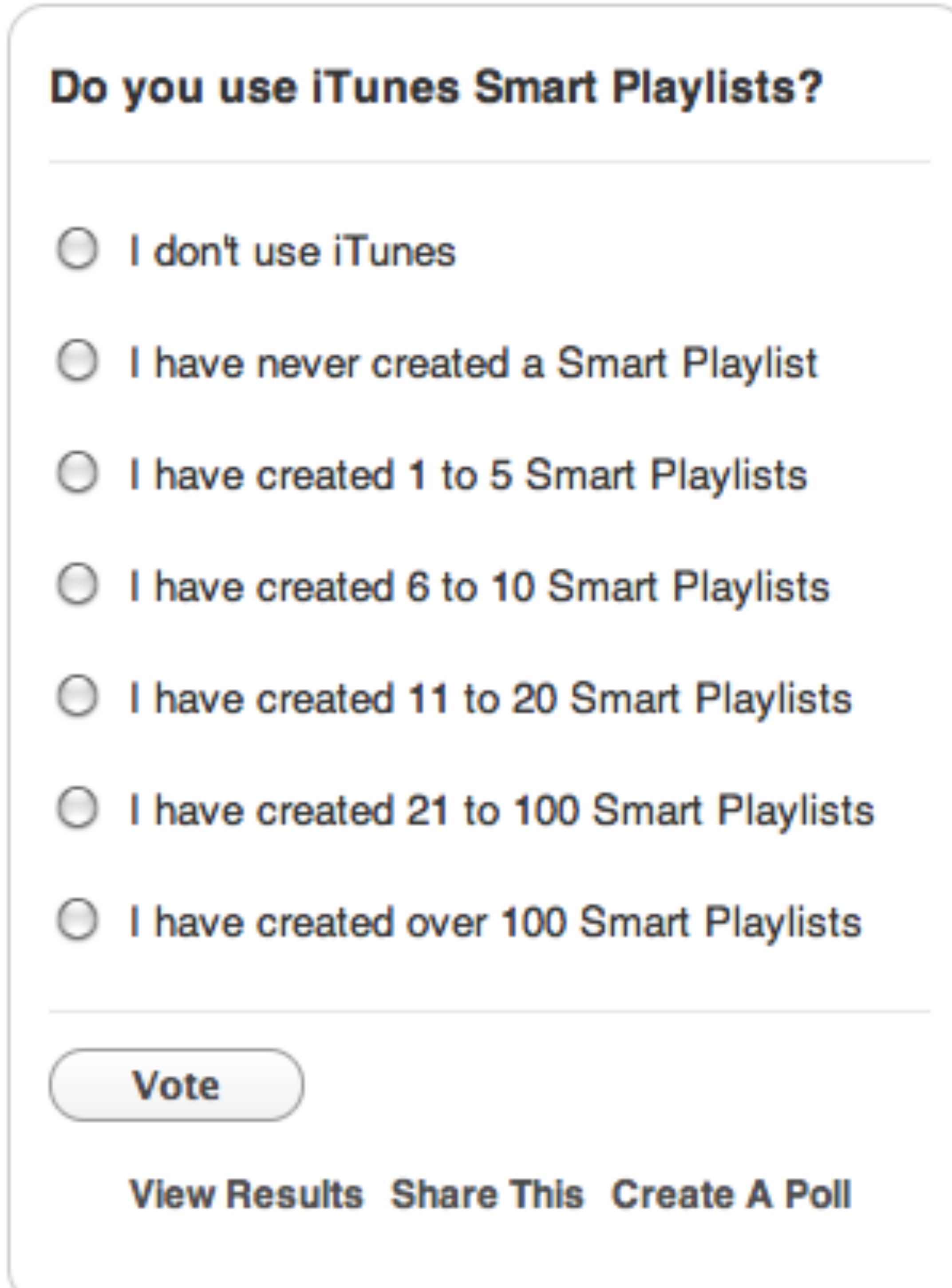
Playlist creation tools



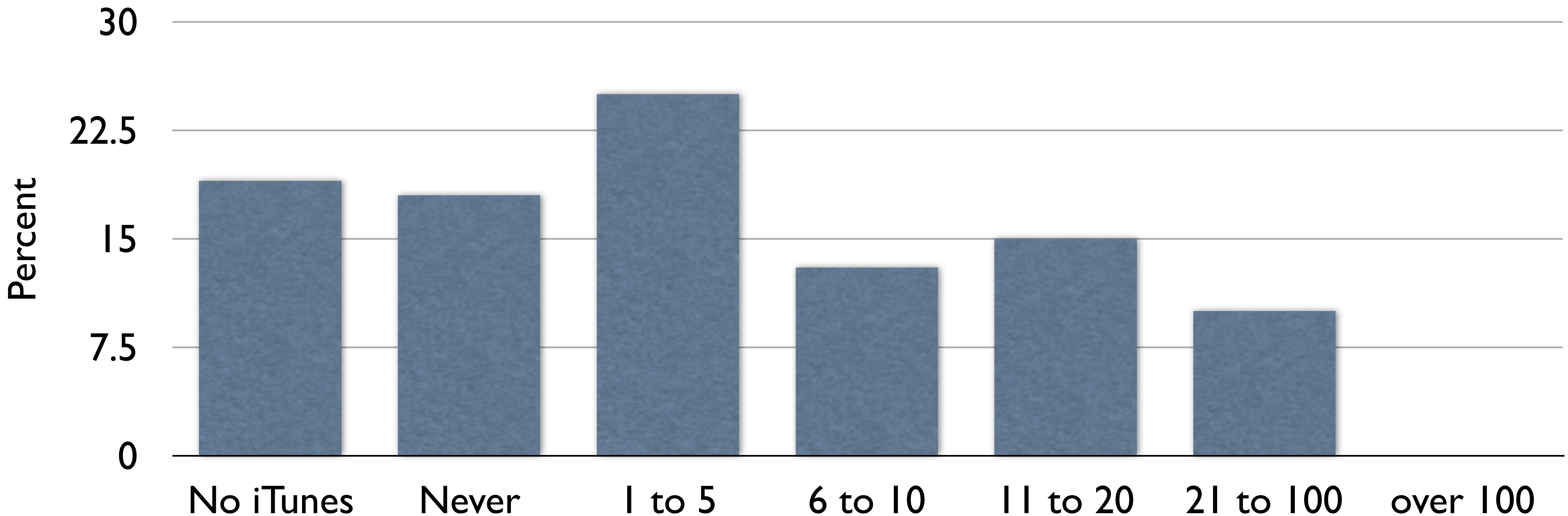
Playlist creation tools



Do people use Smart Playlists?



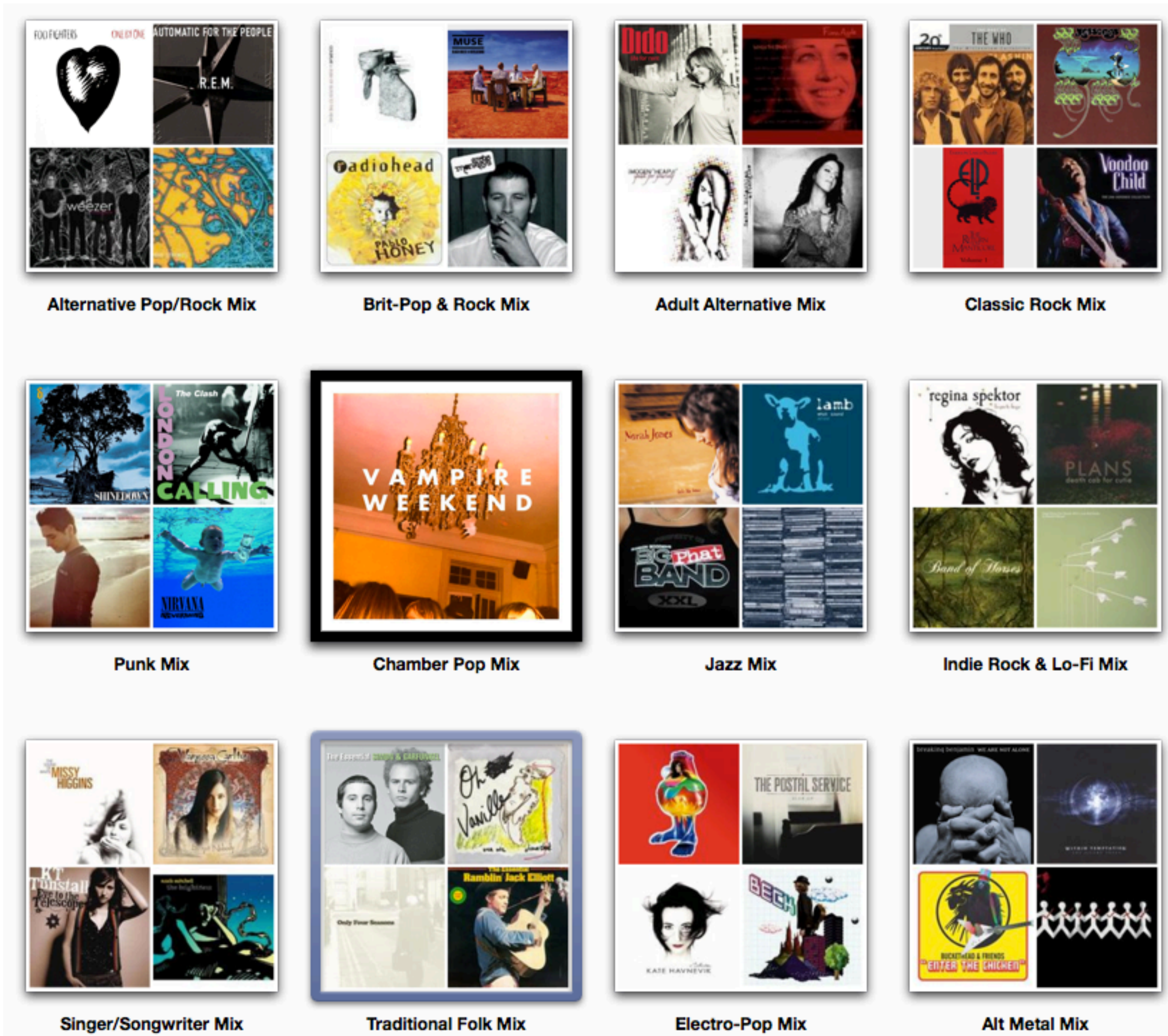
Do people use Smart Playlists?



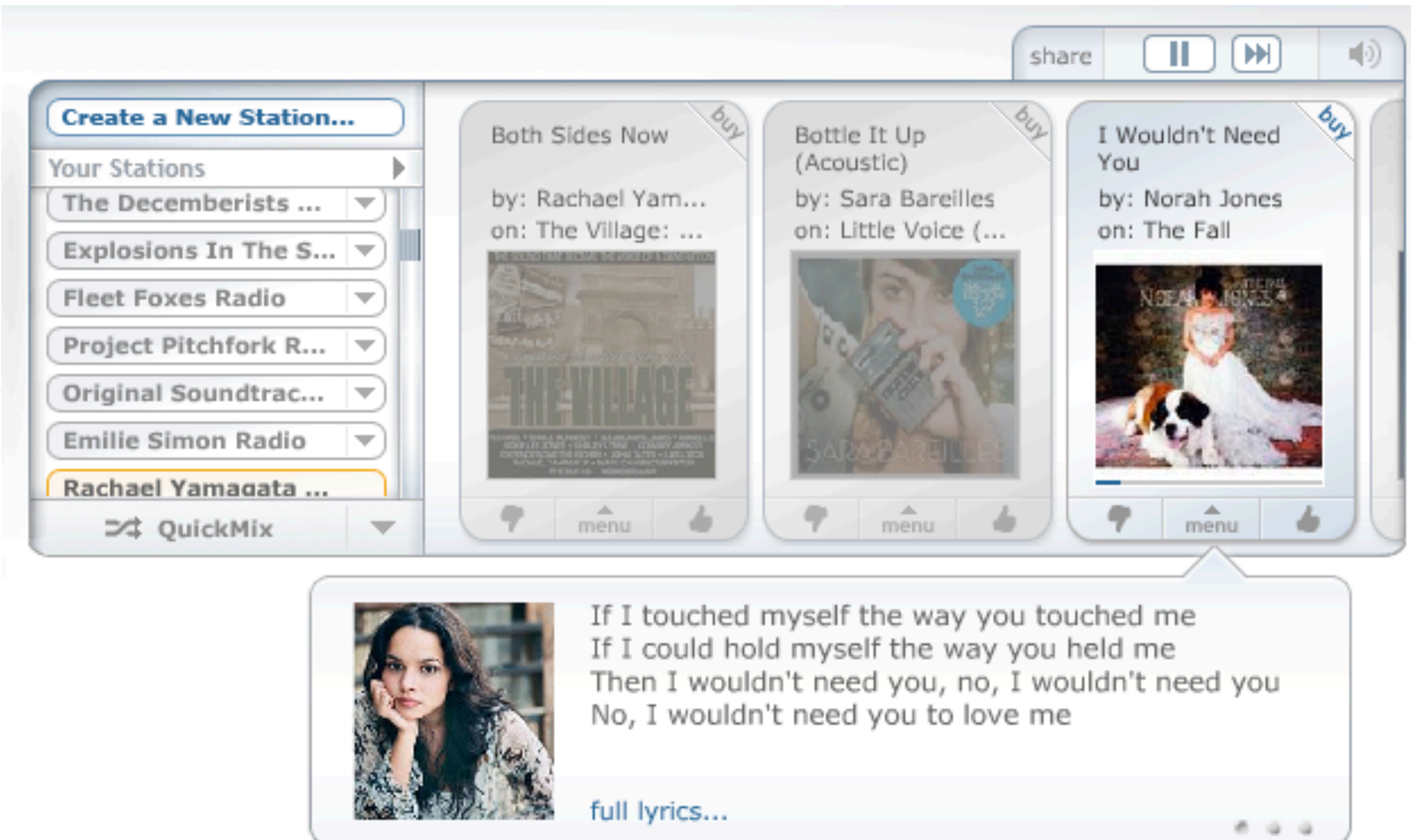
62% of polled have 5 or less smart playlists

Informal poll with 162 respondents

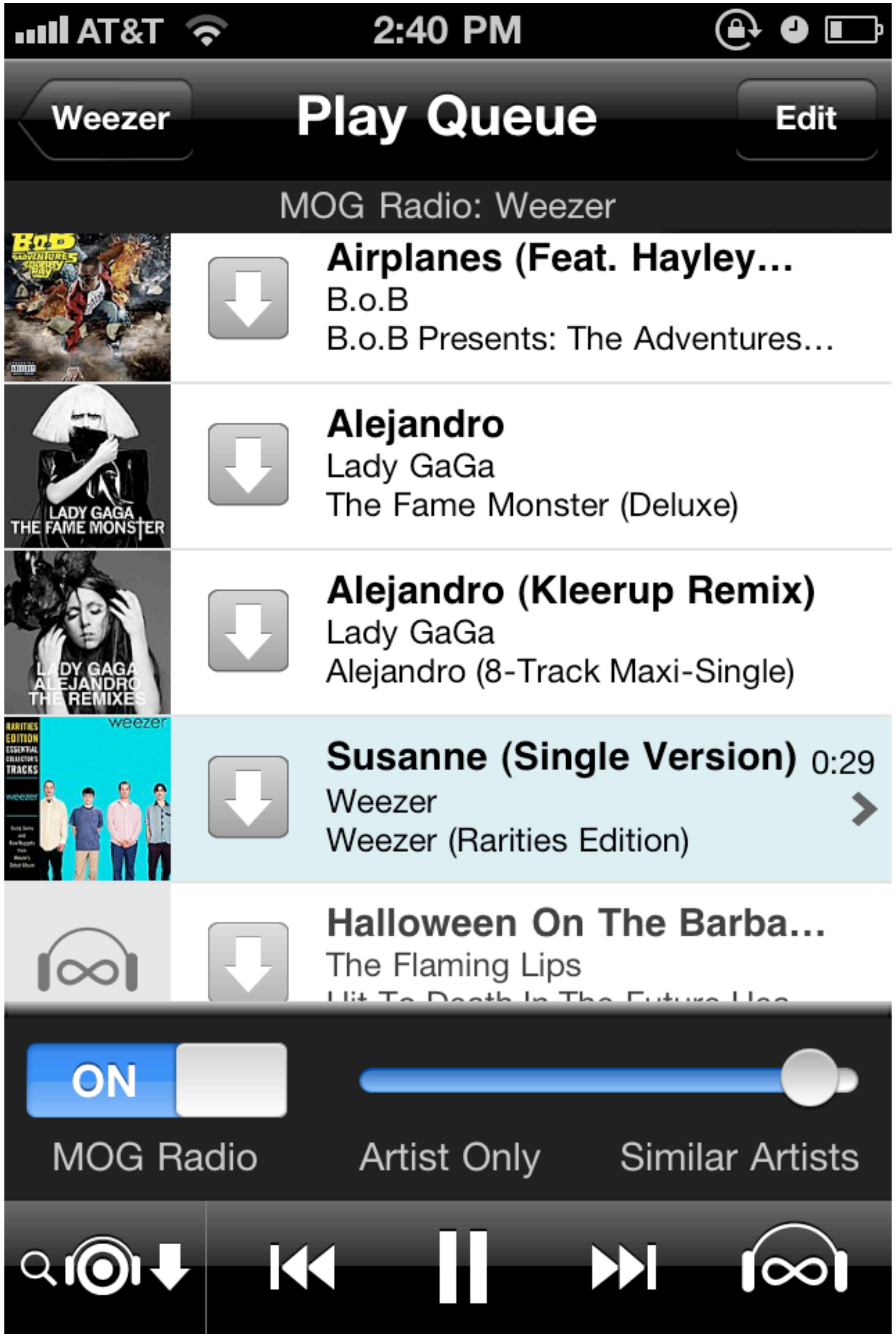
Automated Non-Social iTunes Genius Mix



Automated Non-Social

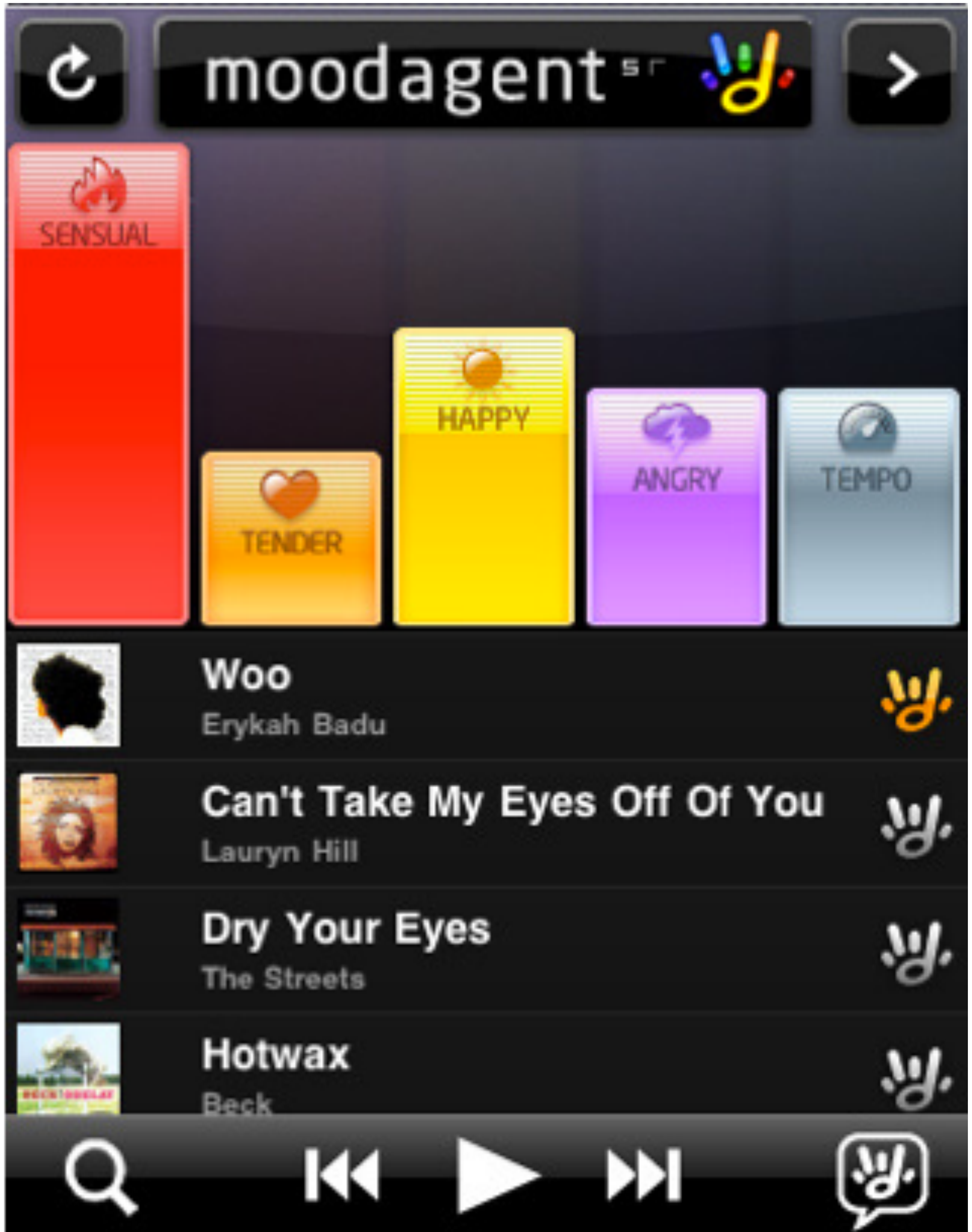


Automated Non-Social



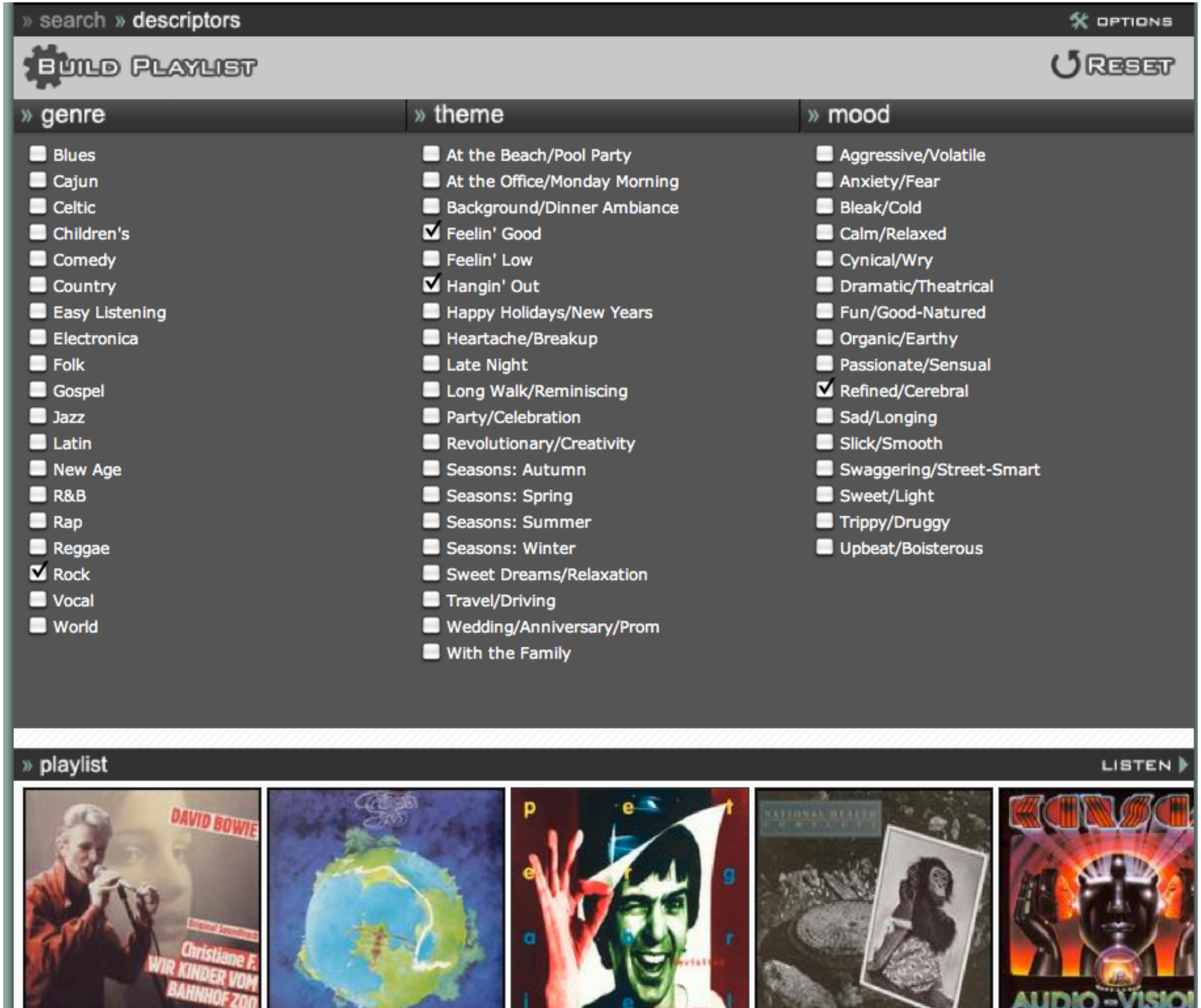
MOG

Mood Agent

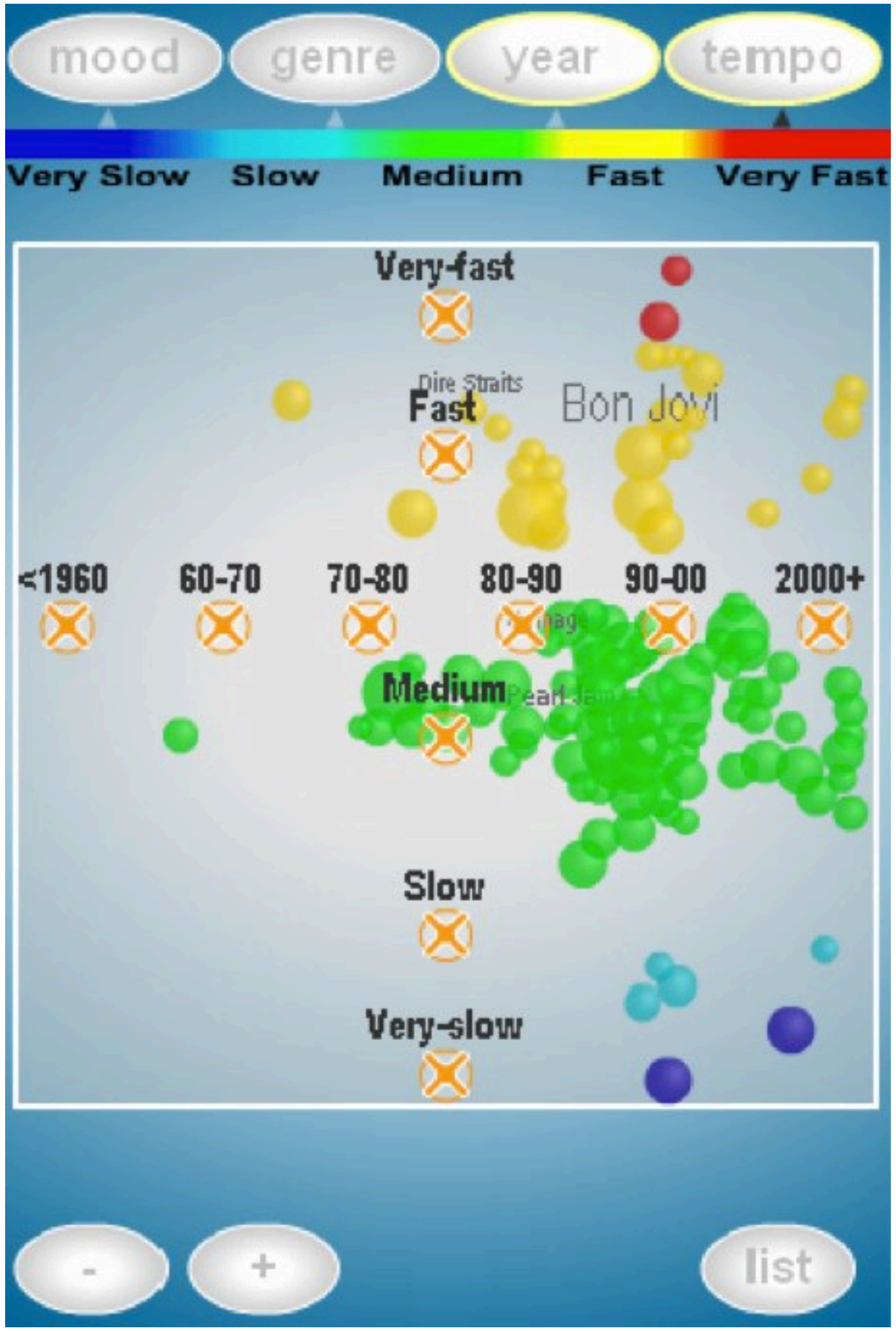


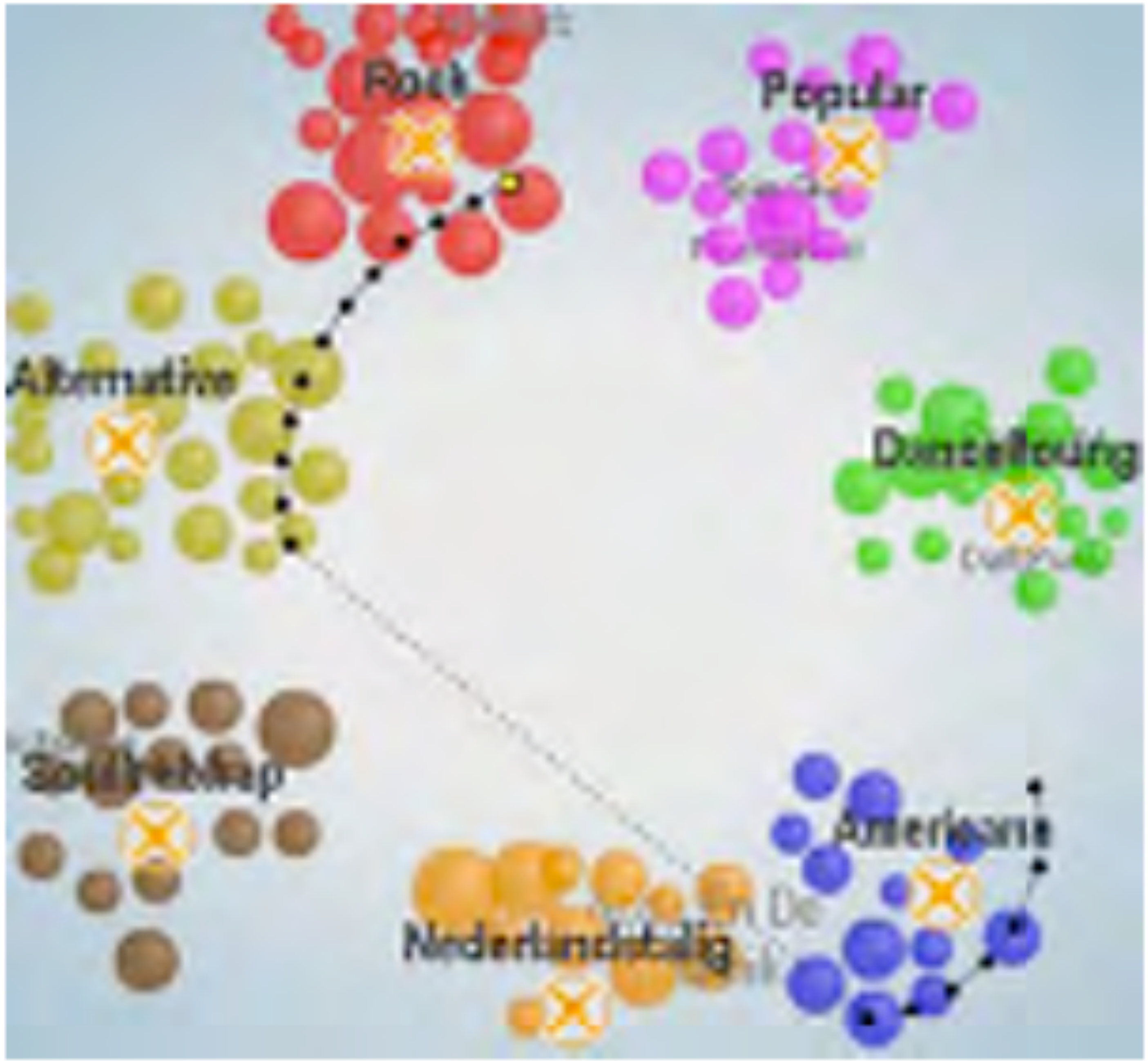
- Use sliders to set levels of 5 'moods':
  - Sensual
  - Tender
  - Happy
  - Angry
  - Tempo

AMG tapestry

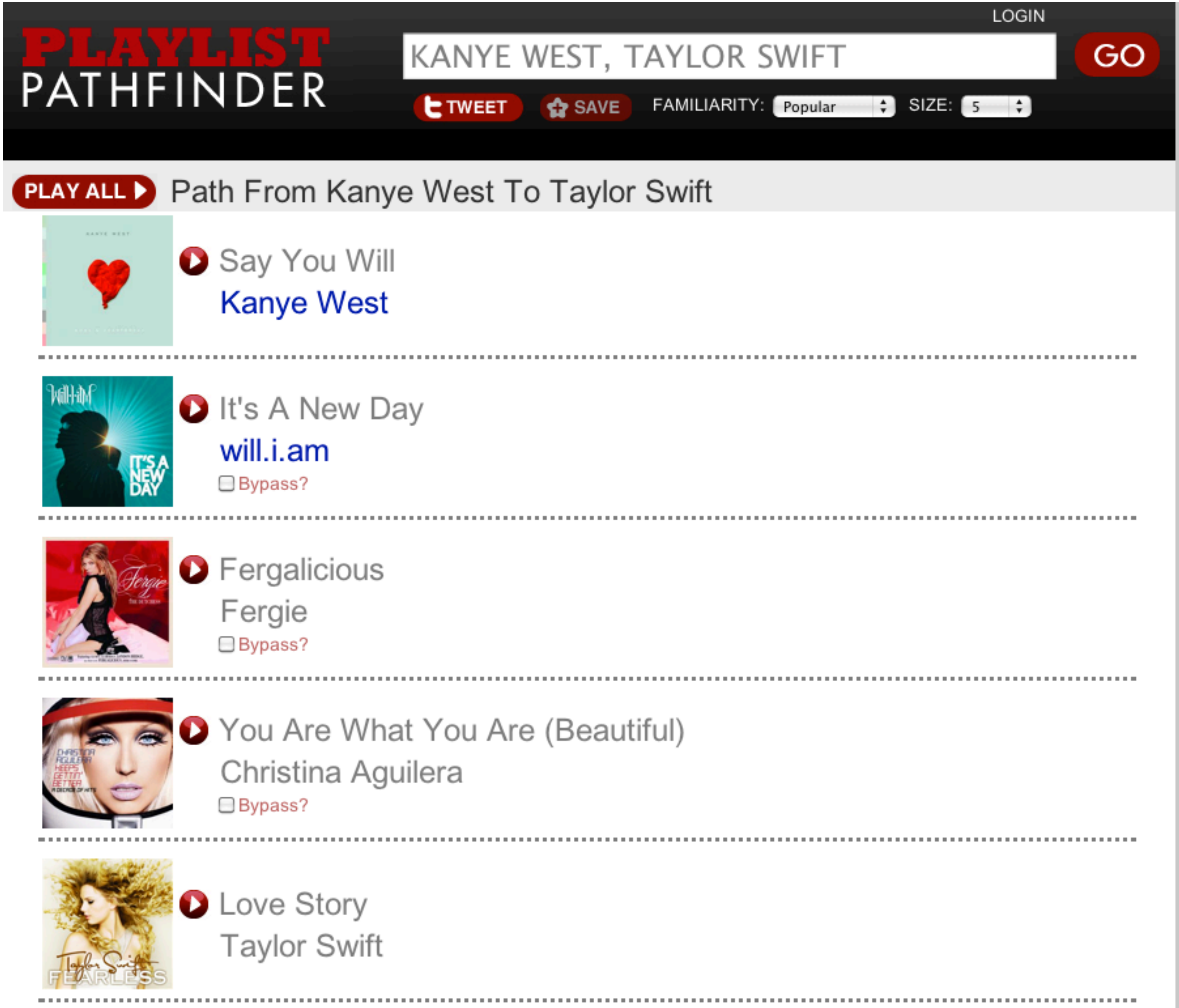


Visual Playlist Generation on the Artist Map

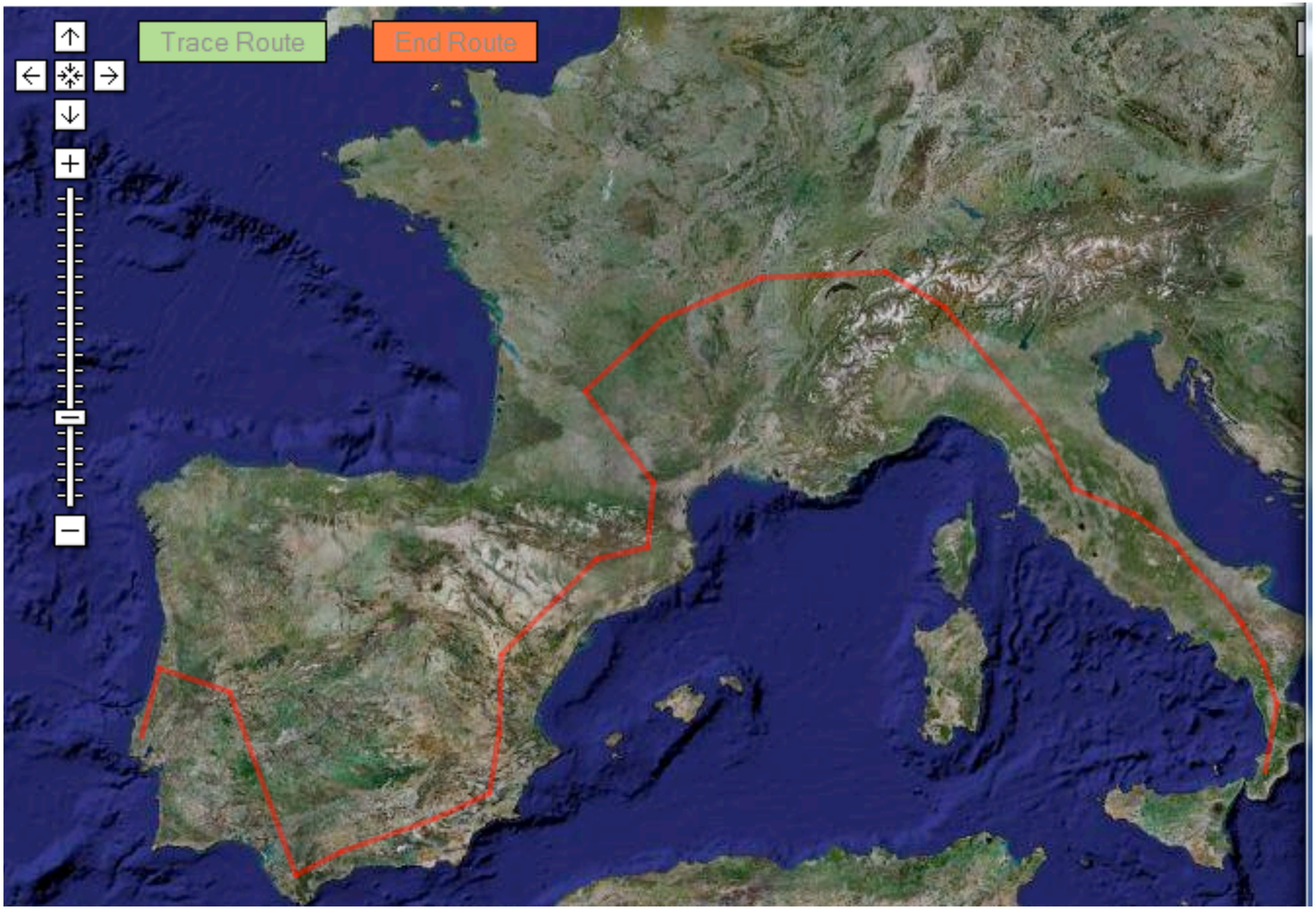




Visual Playlist Generation on the Artist Map
Van Gulick, Vignoli

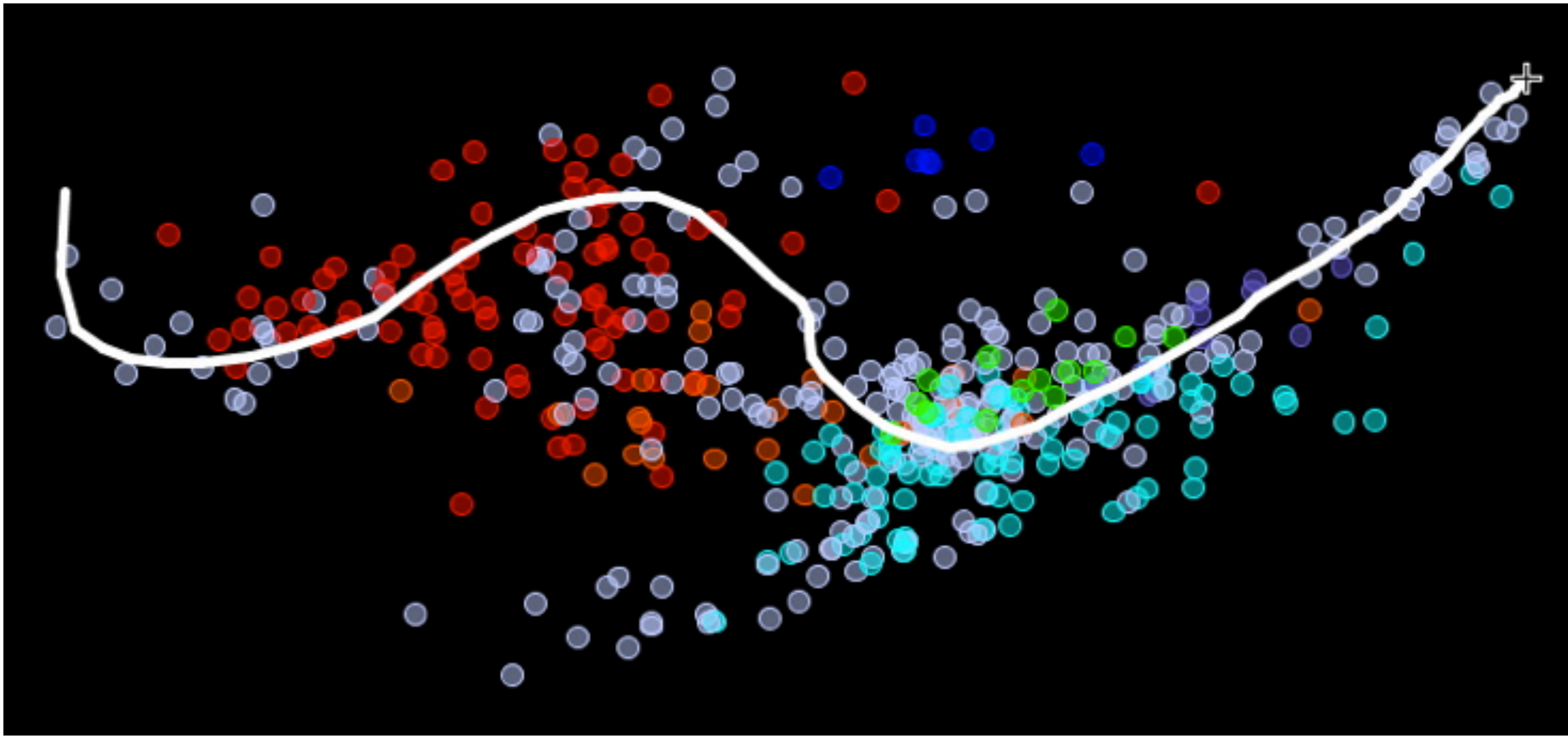

LOGIN
PLAYLIST
PATHFINDER
KANYE WEST, TAYLOR SWIFT
GO
TWEET
SAVE
FAMILIARITY: Popular
SIZE: 5
PLAY ALL
Path From Kanye West To Taylor Swift
Say You Will
Kanye West
It's A New Day
will.i.am
Bypass?
Fergalicious
Fergie
Bypass?
You Are What You Are (Beautiful)
Christina Aguilera
Bypass?
Love Story
Taylor Swift

GeoMuzik



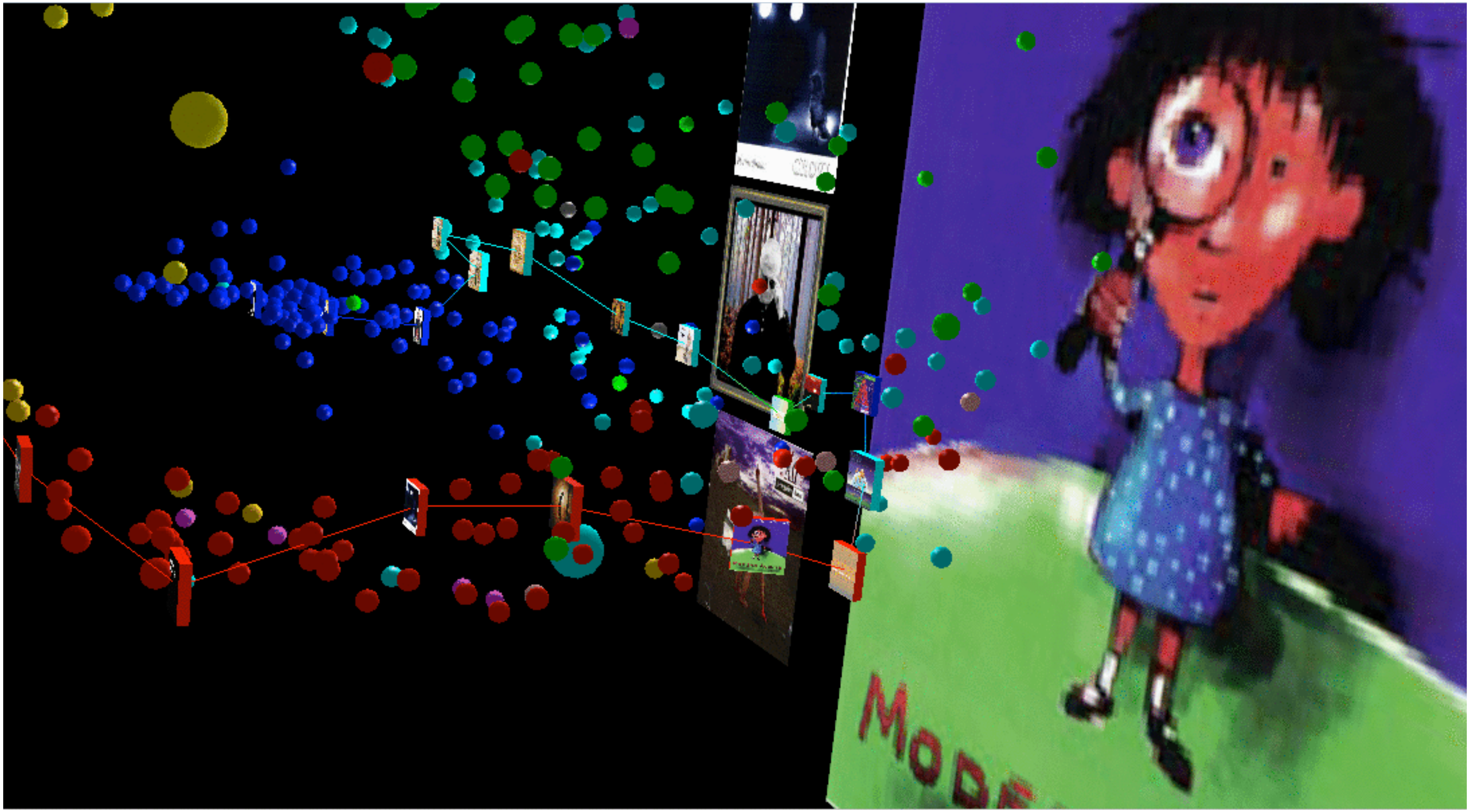
GeoMuzik: A geographic interface for large music collections: Òscar Celma, Marcelo Nunes

Using visualizations to build playlists



MusicBox: Mapping and visualizing music collections
Anita Lillie's Masters Thesis at the MIT Media Lab

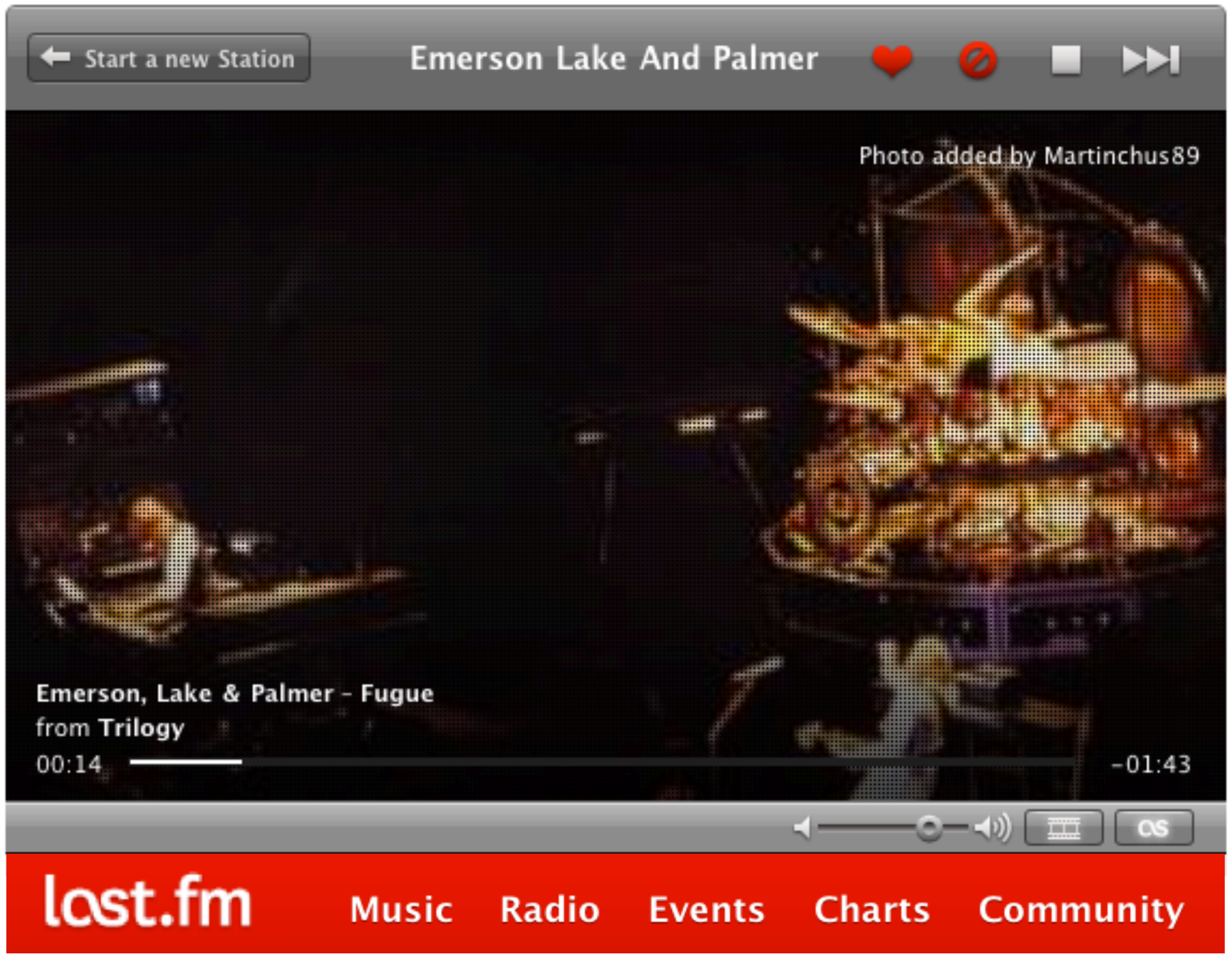
Search Inside the Music

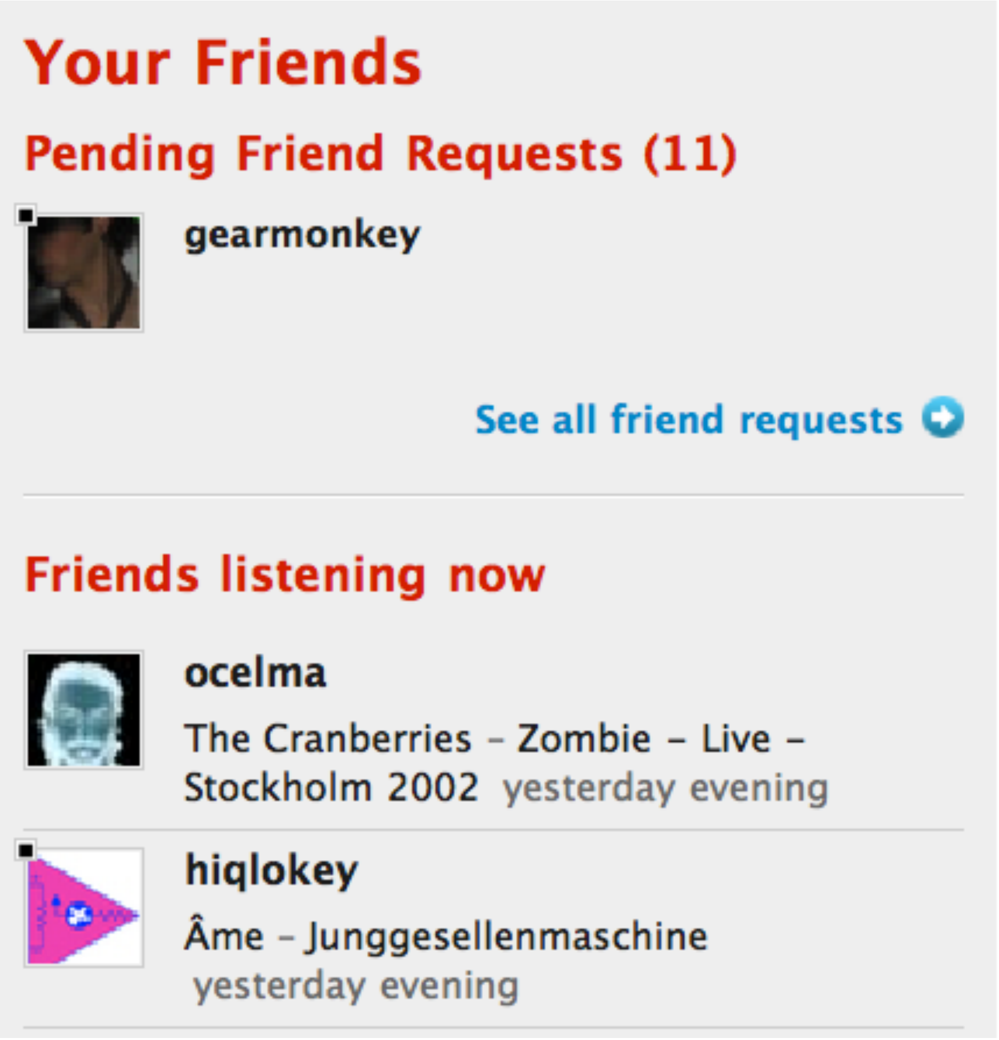


Using 3D visualizations to explore and discover music.
Paul Lamere and Doug Eck


Social
Spotify
playlist.com
RADIO PARADISE
Listener Supported
partyStrands
BETA
iLike
mixpod
last.fm
Manual
Automated
PANDORA
Non-Social

Automated Social
Last.fm





Automated Social

Radio Paradise

DMCA Radio

US rules for Internet streaming radio

- In a single **3 hour period**:
  - No more than **three songs** from the same recording
  - No more than **two songs in a row**, from the same recording
  - No more than **four songs** from the same artist or anthology
  - No more than **three songs in a row** from the same artist or anthology

Note that there are no explicit rules that limit skipping

Terrestrial Radio Programming

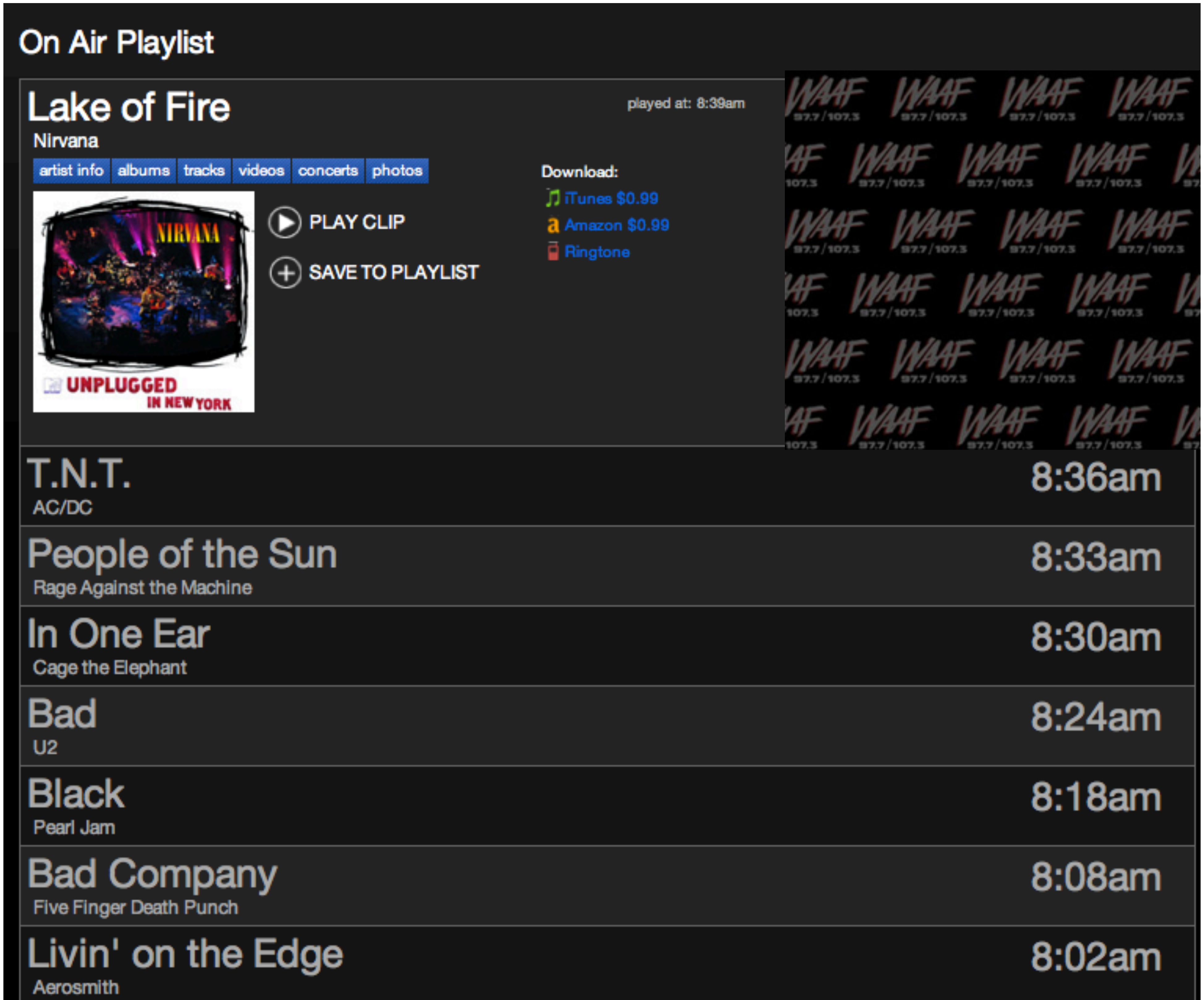
RCS Sound Software®

WFMU
91.1 90.1 fm
wfmu.org

BBC

CLEARCHANNEL RADIO

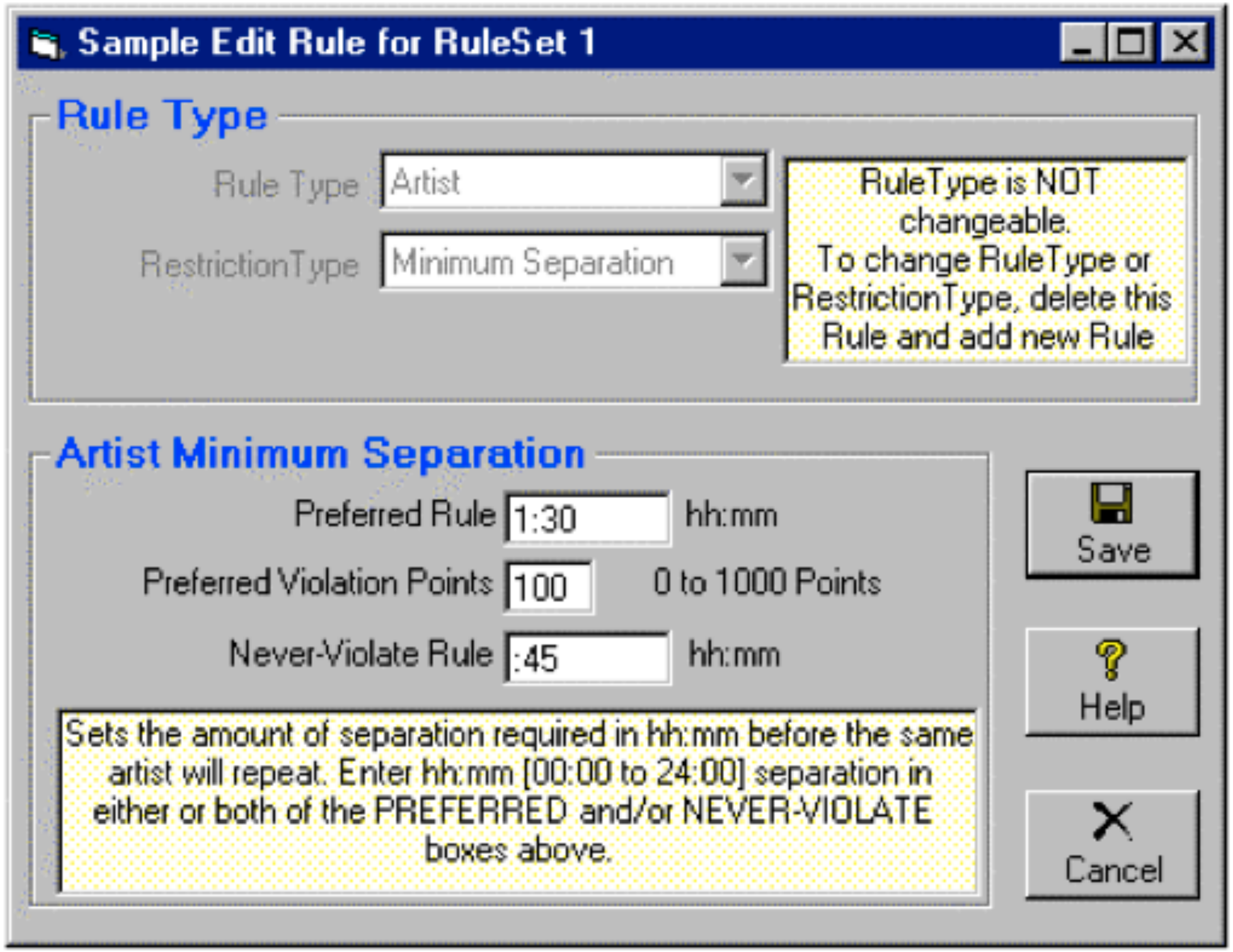
Terrestrial Radio Programming



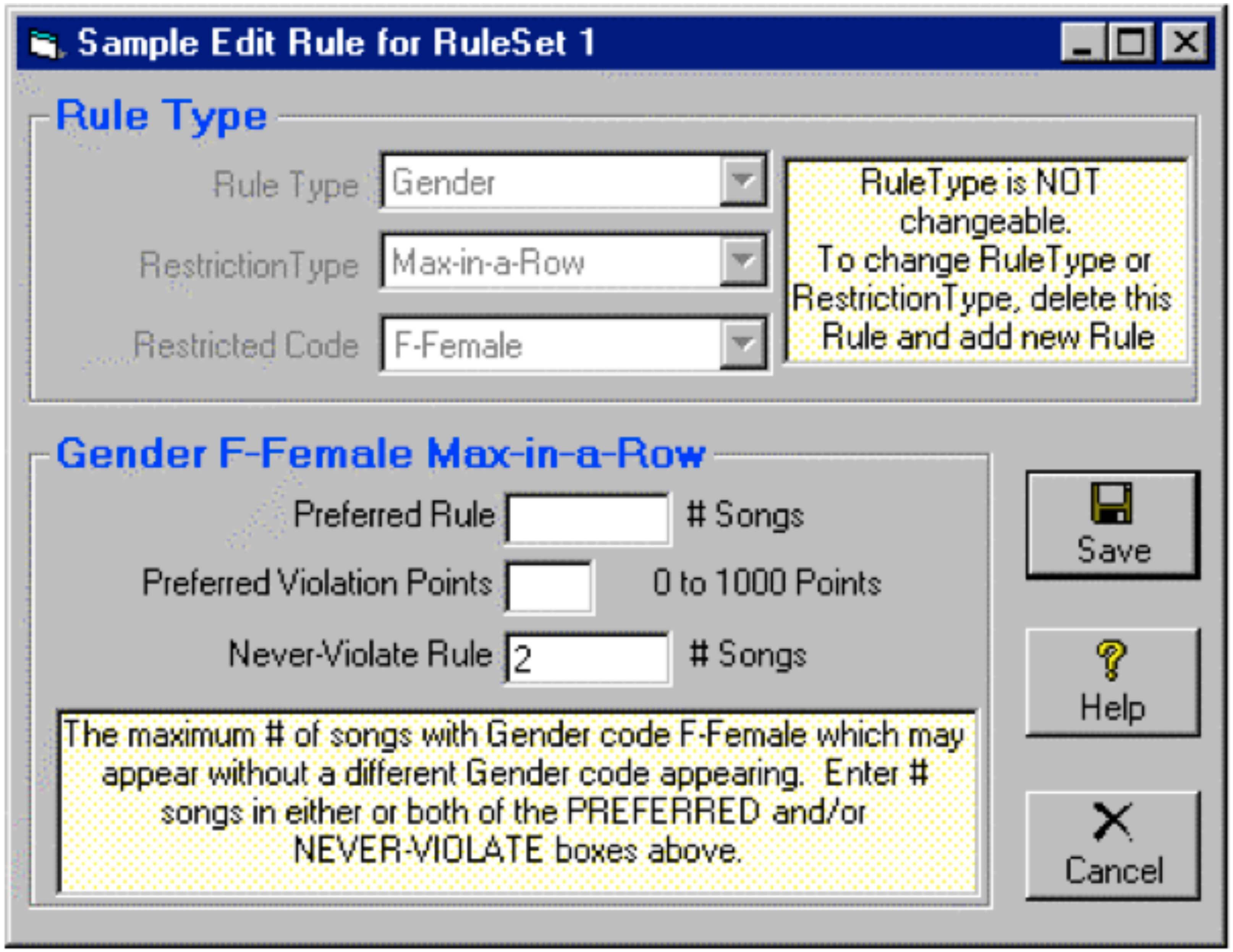
Radio station programming rules

- Divide the day into a set of 5 (typically) 'dayparts'.:
  Mid-6A, 6A-10A, 10A-3P, 3P-7P, and 7P-12Mid
- For each daypart:
  - Gender, Tempo, Intensity, Mood, Style controls
  - Artist separation controls [global and individual artist]
  - Prior-day horizontal title separation
  - Artist blocks [multiple songs in-a-row by same artist]
  - "Never-Violate" and "Preferred" rules
  - Hour circulation rules

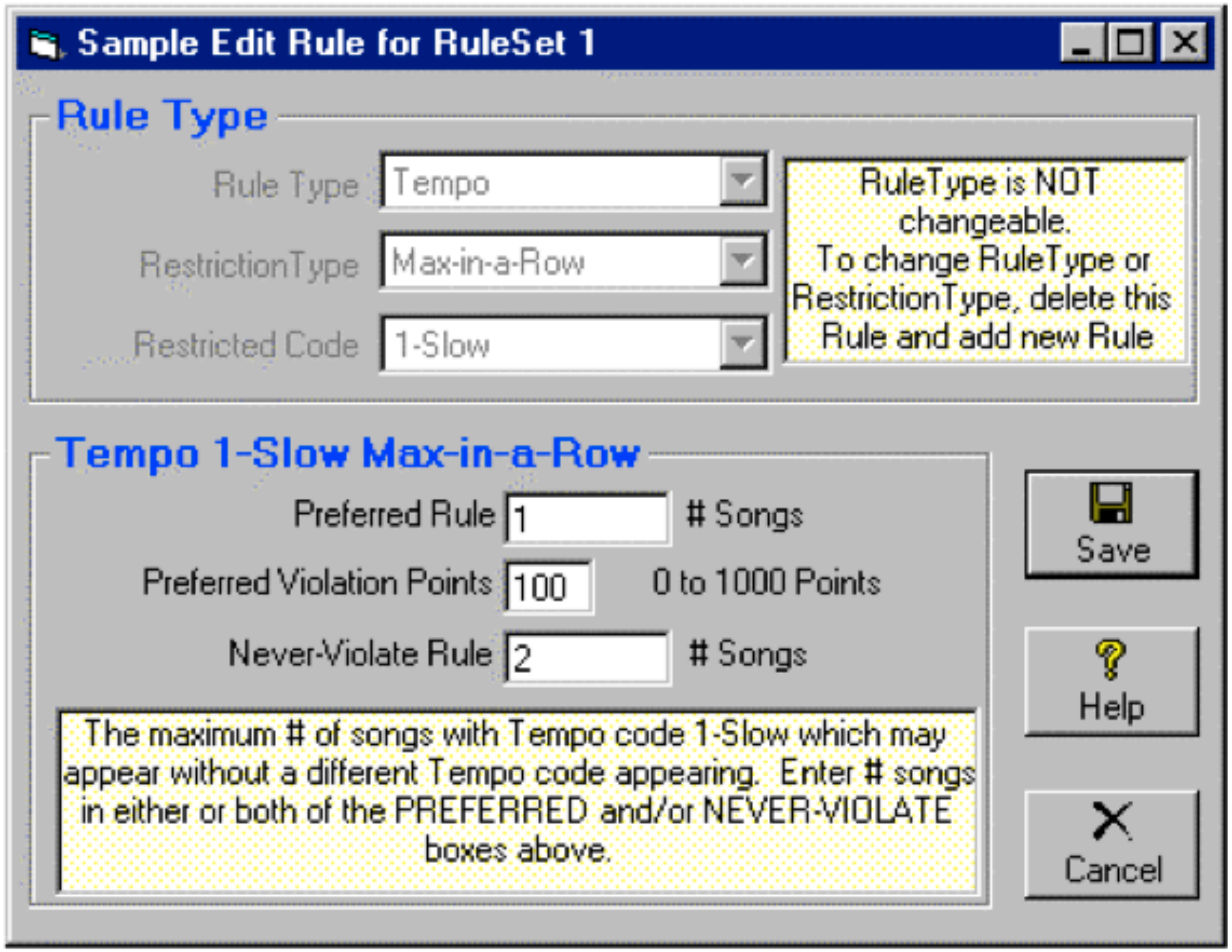
Automated Radio Programming



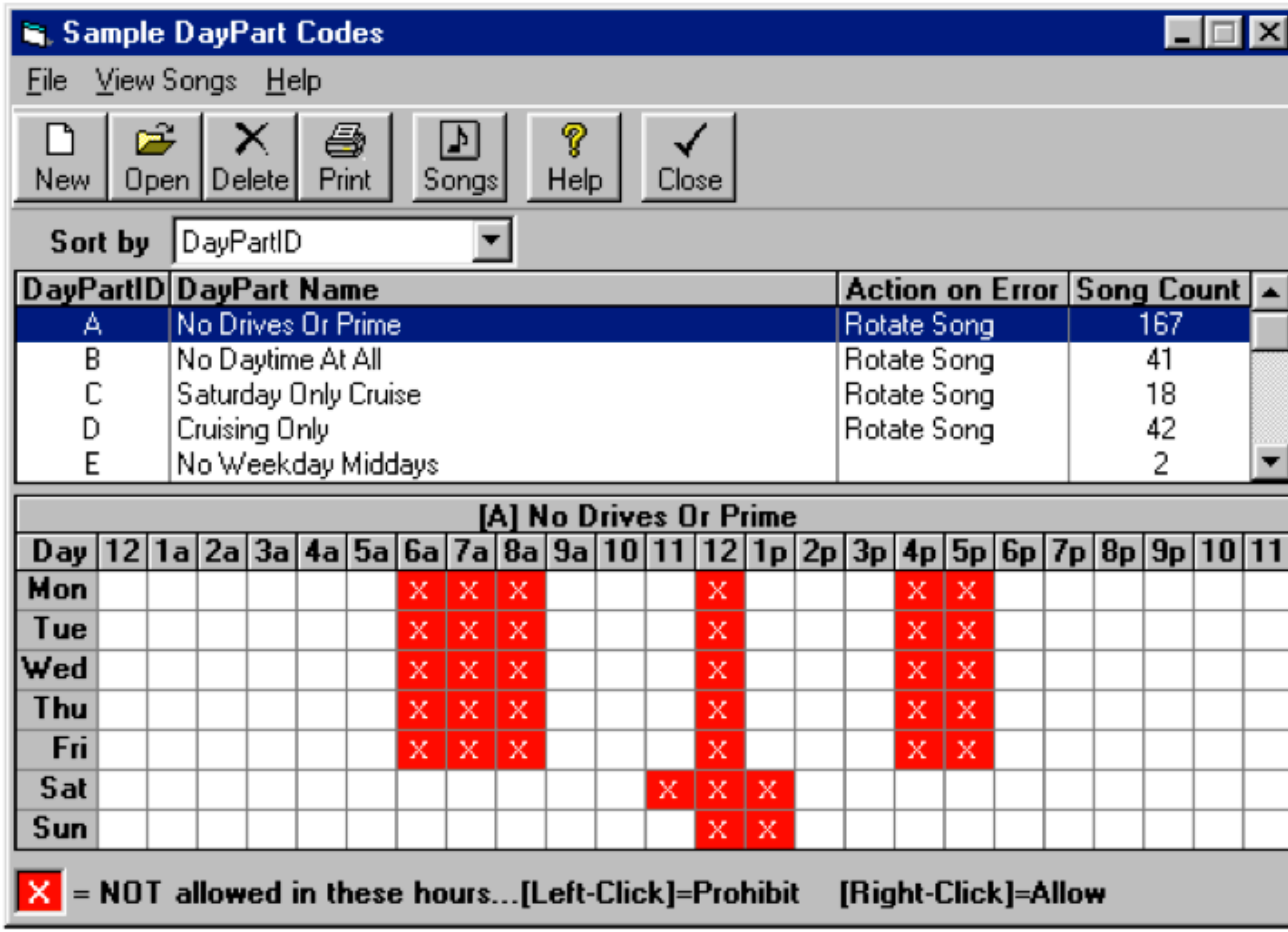
Automated Radio Programming



Automated Radio Programming



Automated Radio Programming



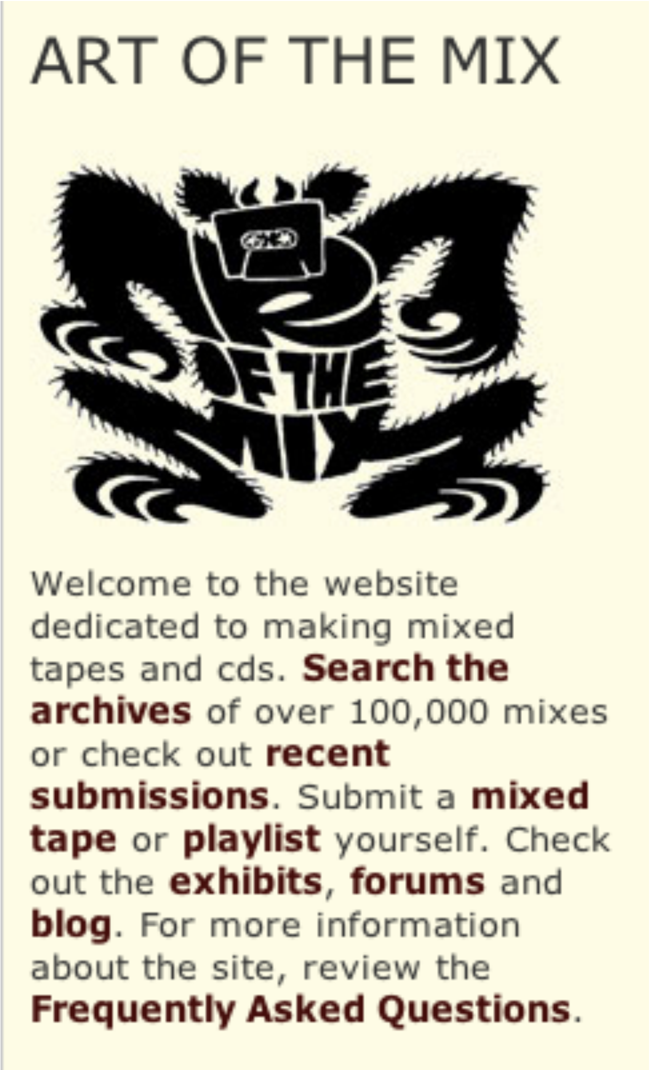
Automated Social PartyStrands


Social
Spotify
playlist.com
RADIO PARADISE
partyStrands
iLike
mixpod
last.fm
Manual
Automated
PANDORA
Non-Social

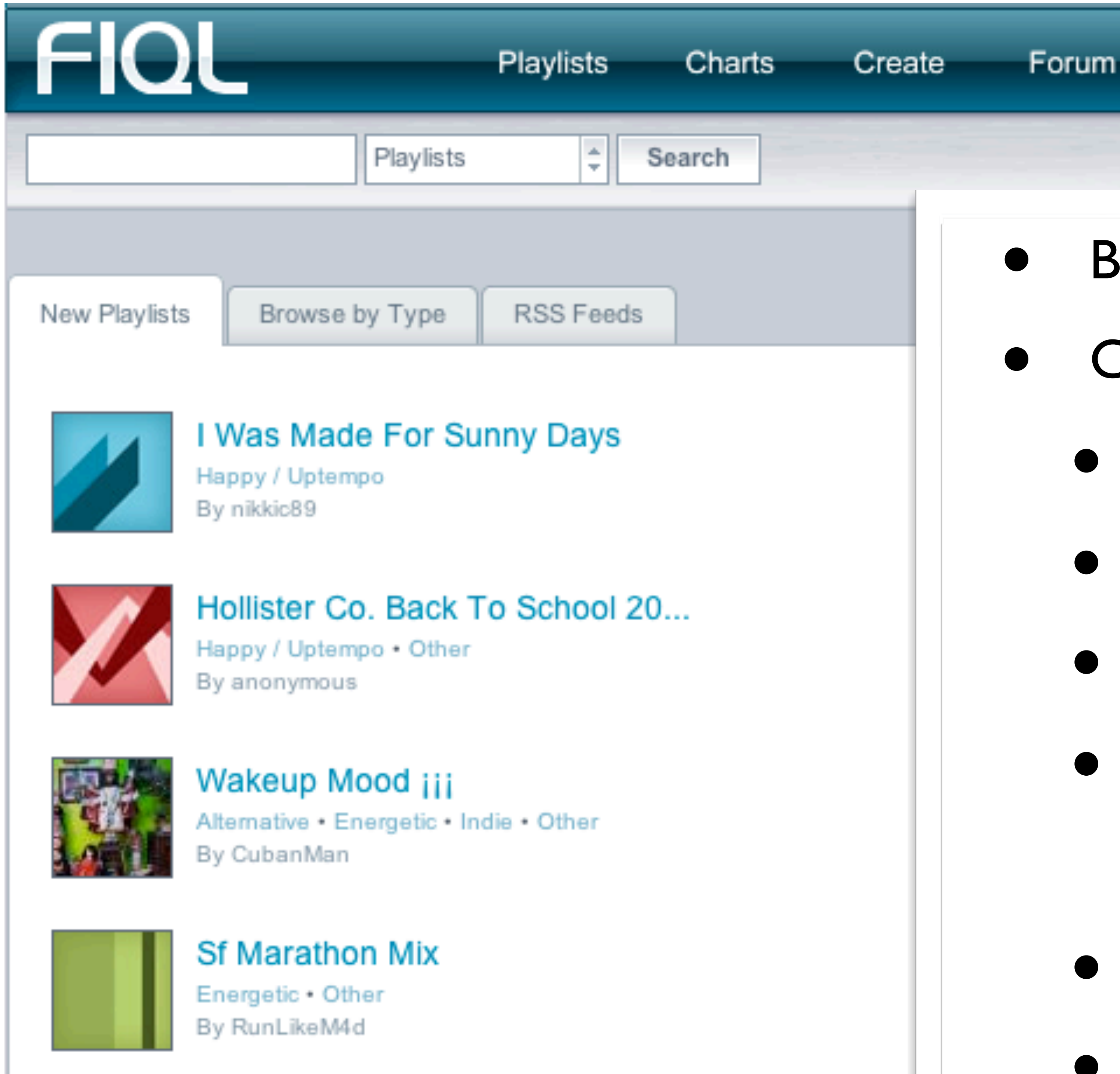
art of the mix

- Hand made playlists
- Mix art
- Web services
- Pre-crawled data at:

http://labrosa.ee.columbia.edu/projects/musicsim/aotm.html

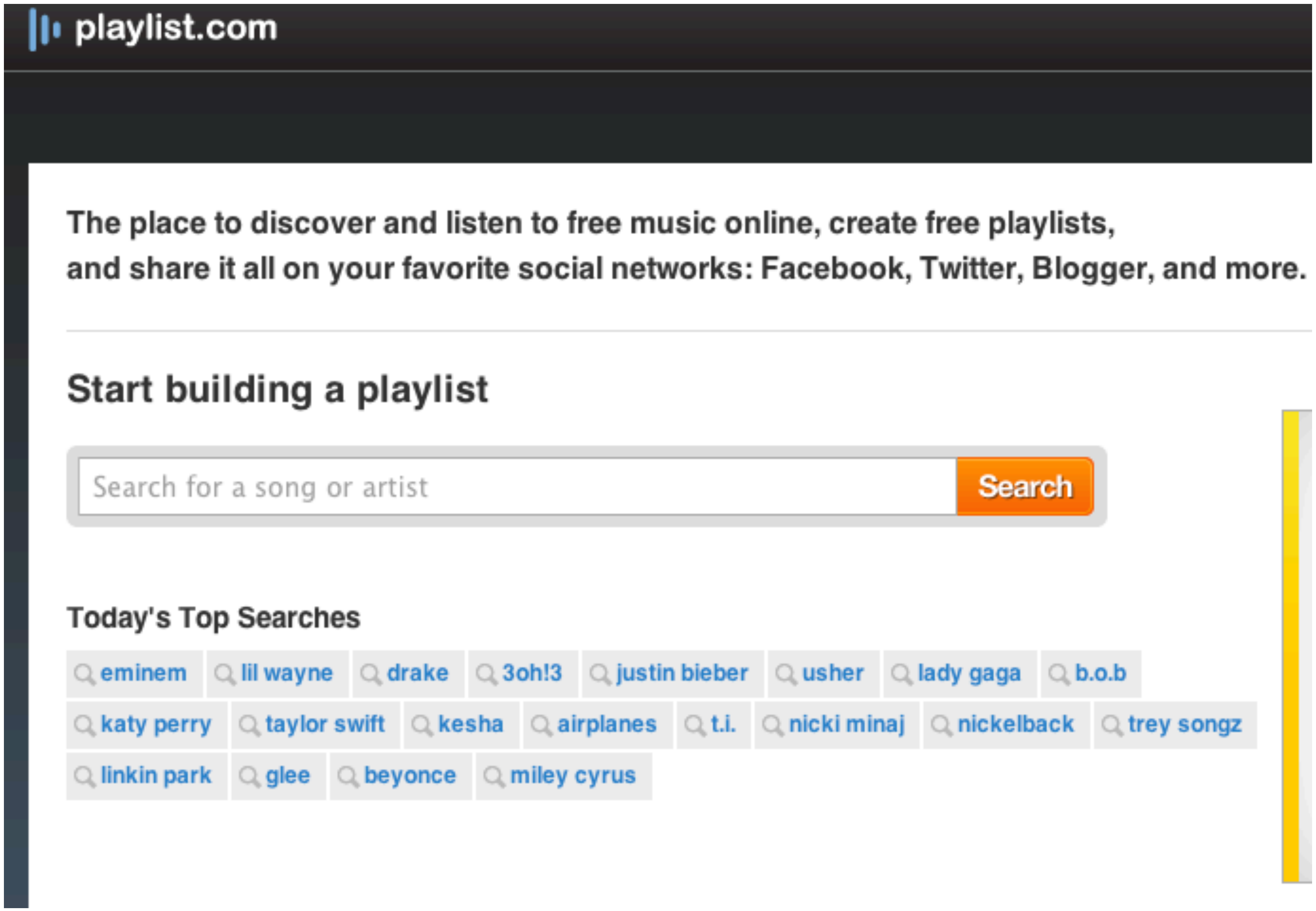


fiql.com



- Browse / search for playlists
- Create a playlist:
  - Search for artist / songs
  - Add songs to a playlist
  - Re-order the playlist
  - Describe the playlist:
    - title, description, tags
  - Decorate the playlist
  - Publish the playlist

Playlist.com



mixpod

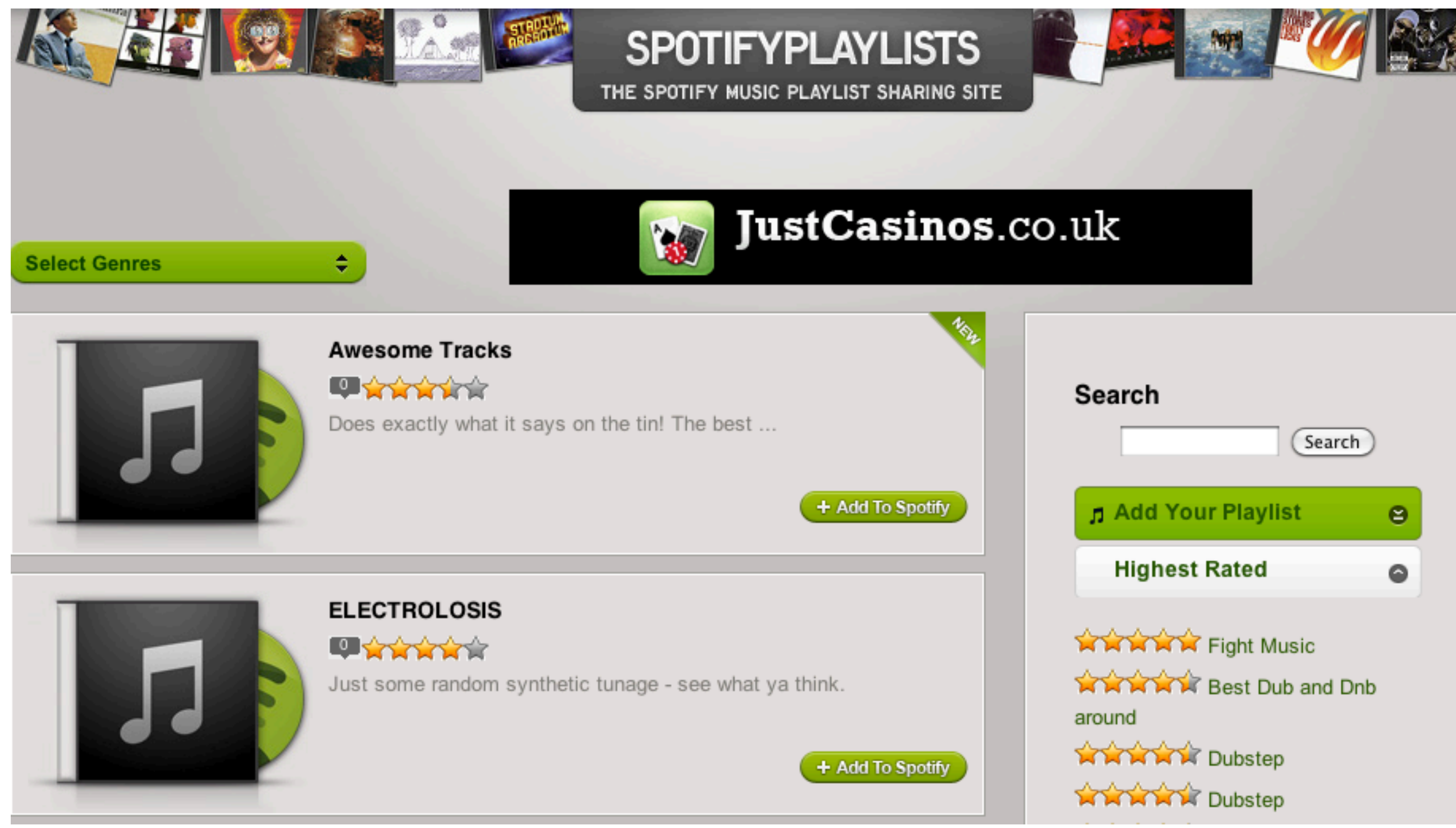

mixpod
My MixPod
My Playlists
Create Playlist
Browse
Community
Create A Music Playlist!
Your music says a lot about you.
Enter an artist, a song title or both...
Search
Popular Searches:
Mike Posner, Katy Perry, Taio Cruz, BoB, Jason Derülo, Eminem, Drake, Eminem, Usher, Travie McCoy
Share your playlist on MySpace, Facebook, Friendster, hi5, Bebo, blogs & more.
myspace bebo friendster imvu orkut hi5 xanga
Sign Up!
Social Playlists
Funk Music (25 tracks)
1. Play That Funky Music - W ...
2. Stevie Wonder "SUPER ...
3. Super Freak By Rick James
4. Brick - Dazz
5. Give Up The Funk By Parli ...
and 20 more
Popular Songs
Justin Bieber - Baby Ft. Ludacris
31,068 listeners
Sean Kingston & Justin Bieber "eenie Meenie"
17,540 listeners
Not Afraid By Eminem

Spotify

- Sharable playlists
- Collaborative playlists
- Many 3rd party playlist sites

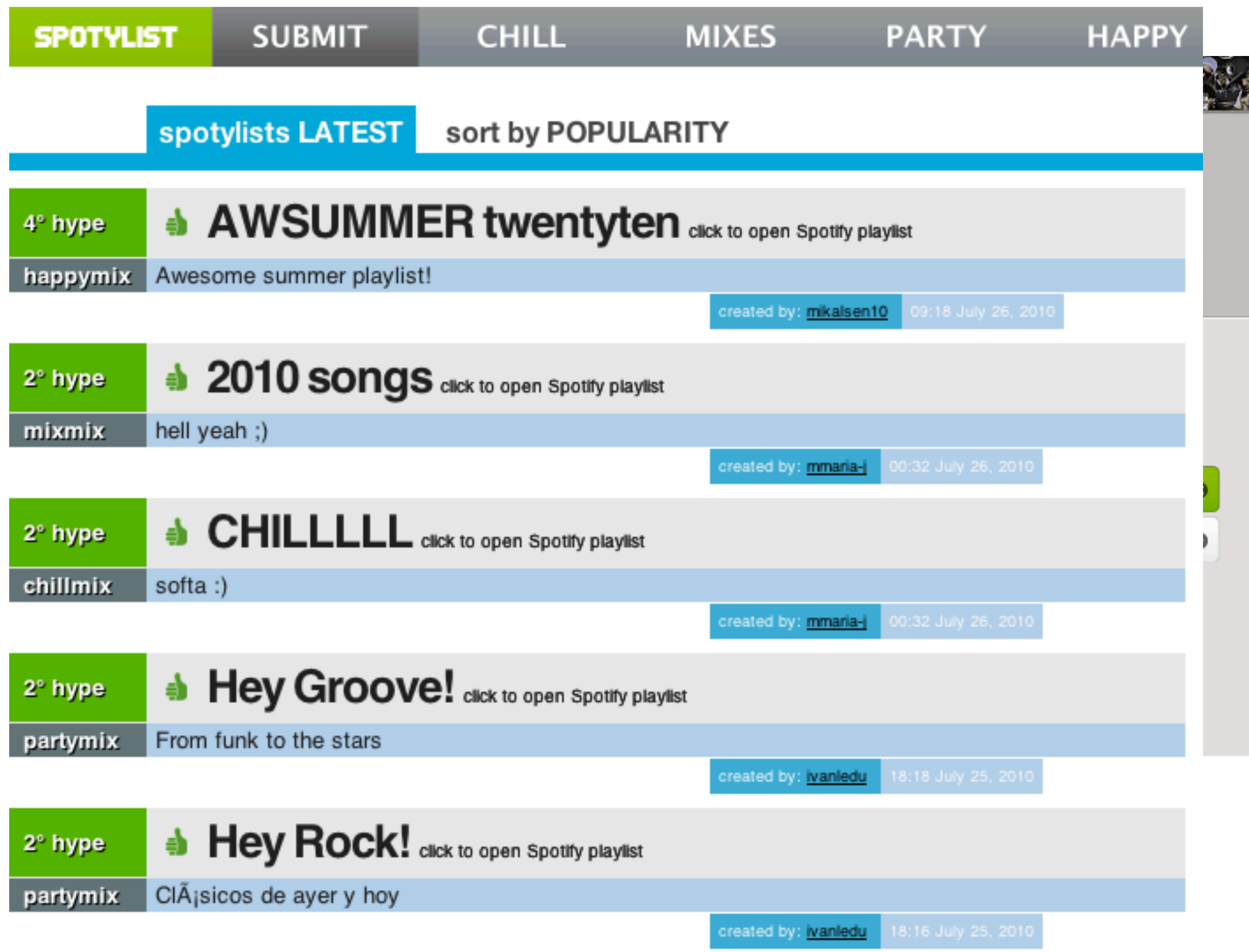
Spotify

- Sharable playlists
- Collaborative playlists
- Many 3rd party playlist sites



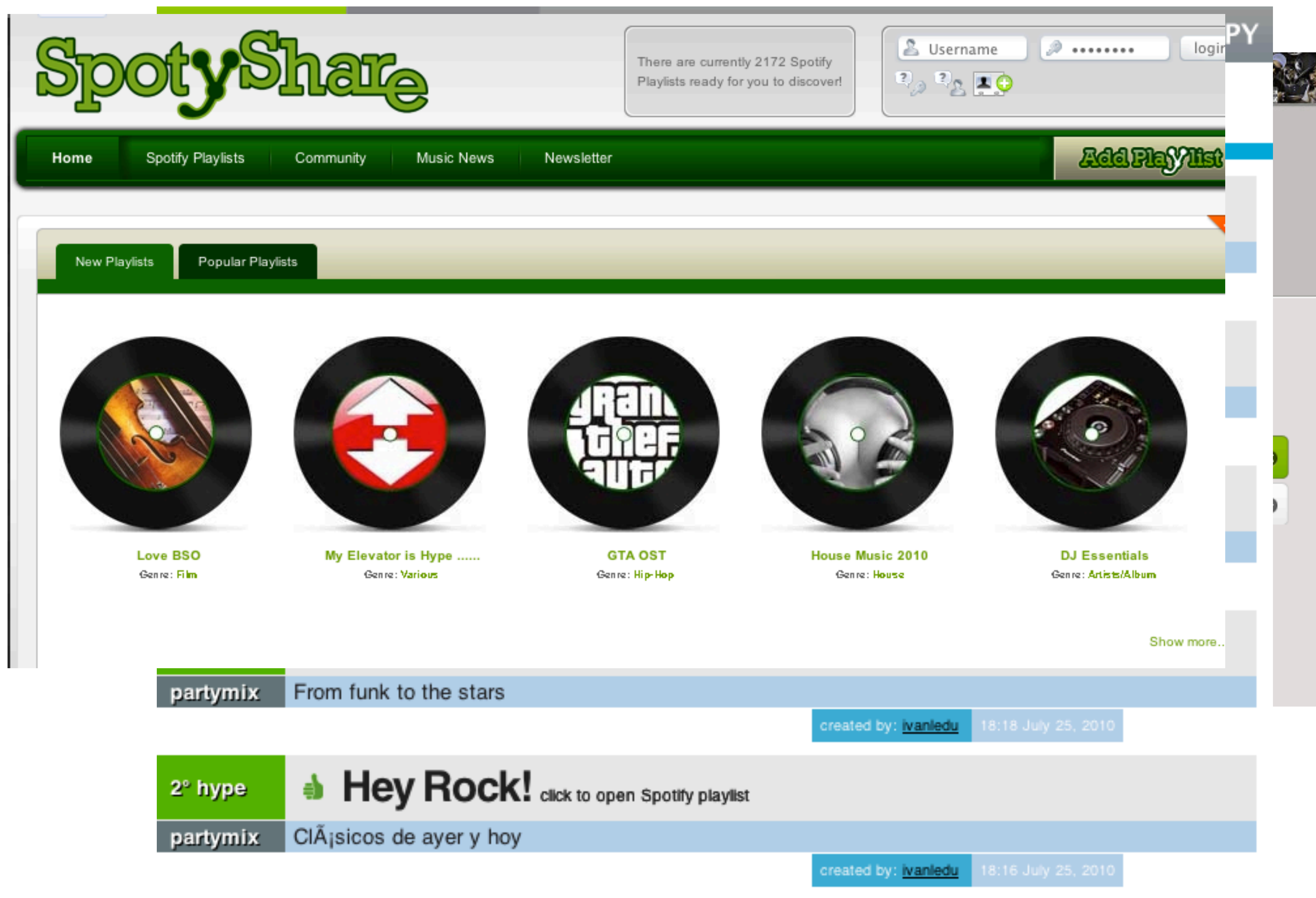
Spotify

- Sharable playlists
- Collaborative playlists
- Many 3rd party playlist sites



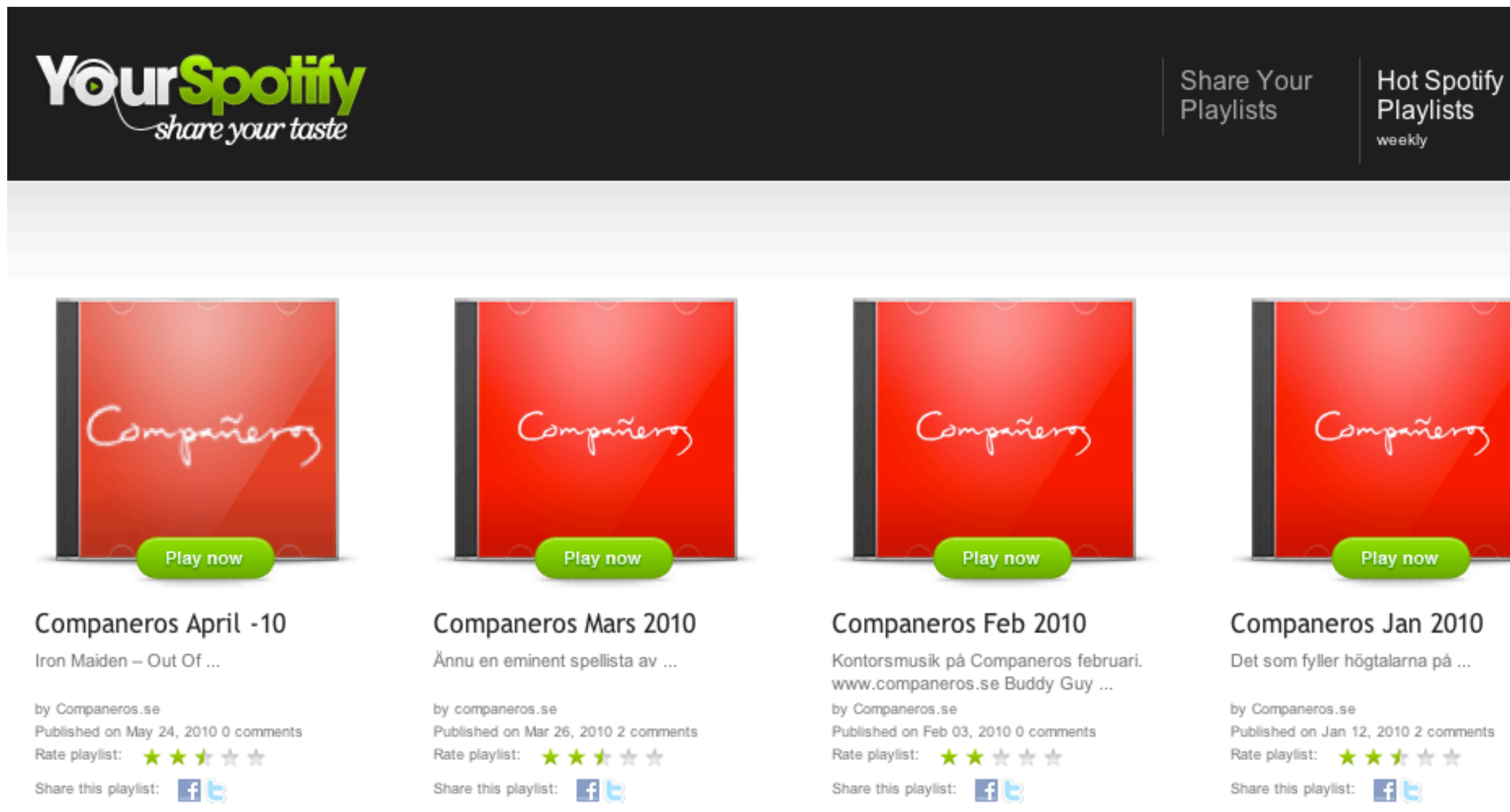
Spotify

- Sharable playlists
- Collaborative playlists
- Many 3rd party playlist sites



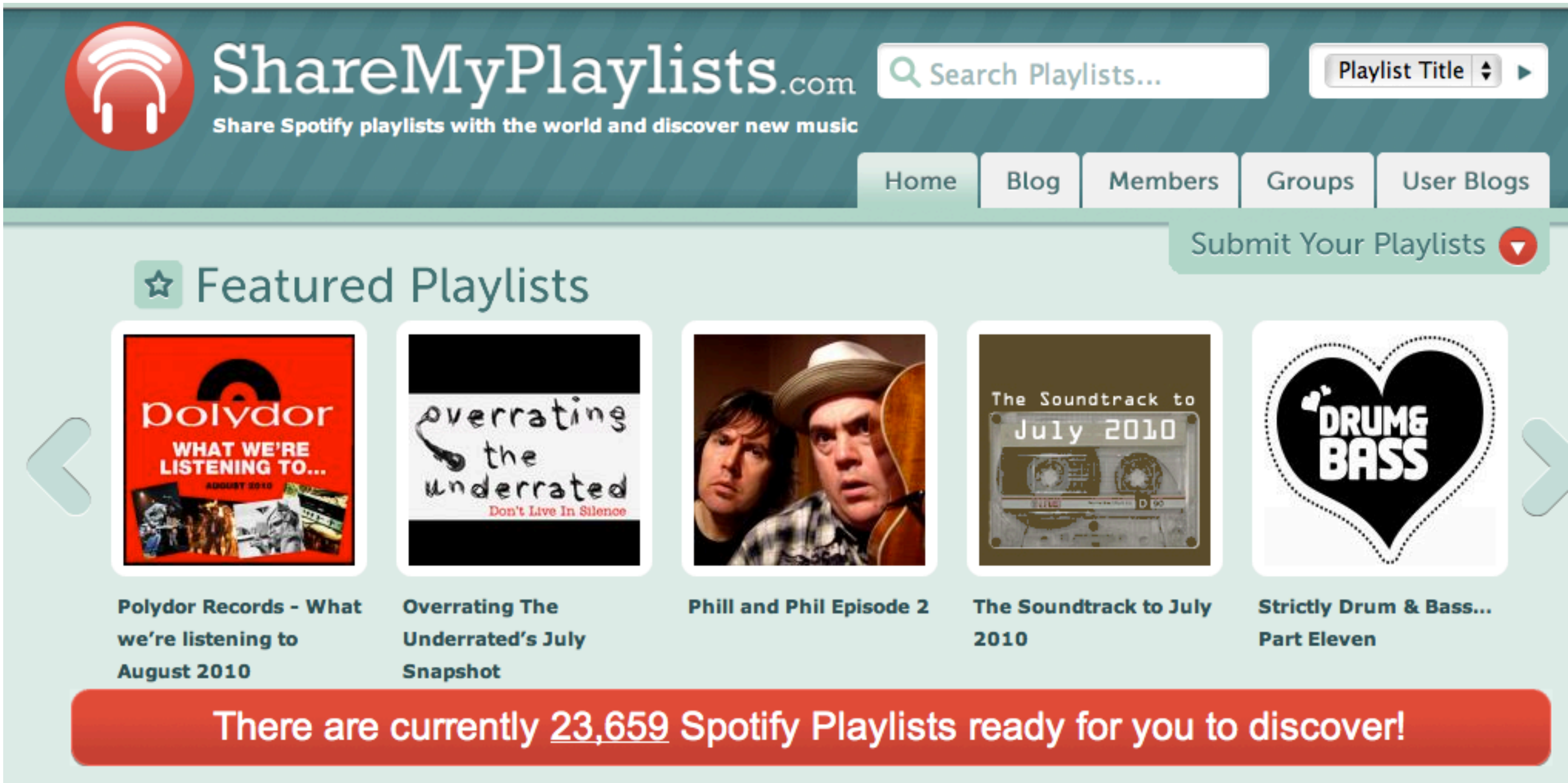
Spotify

- Sharable playlists
- Collaborative playlists
- Many 3rd party playlist sites



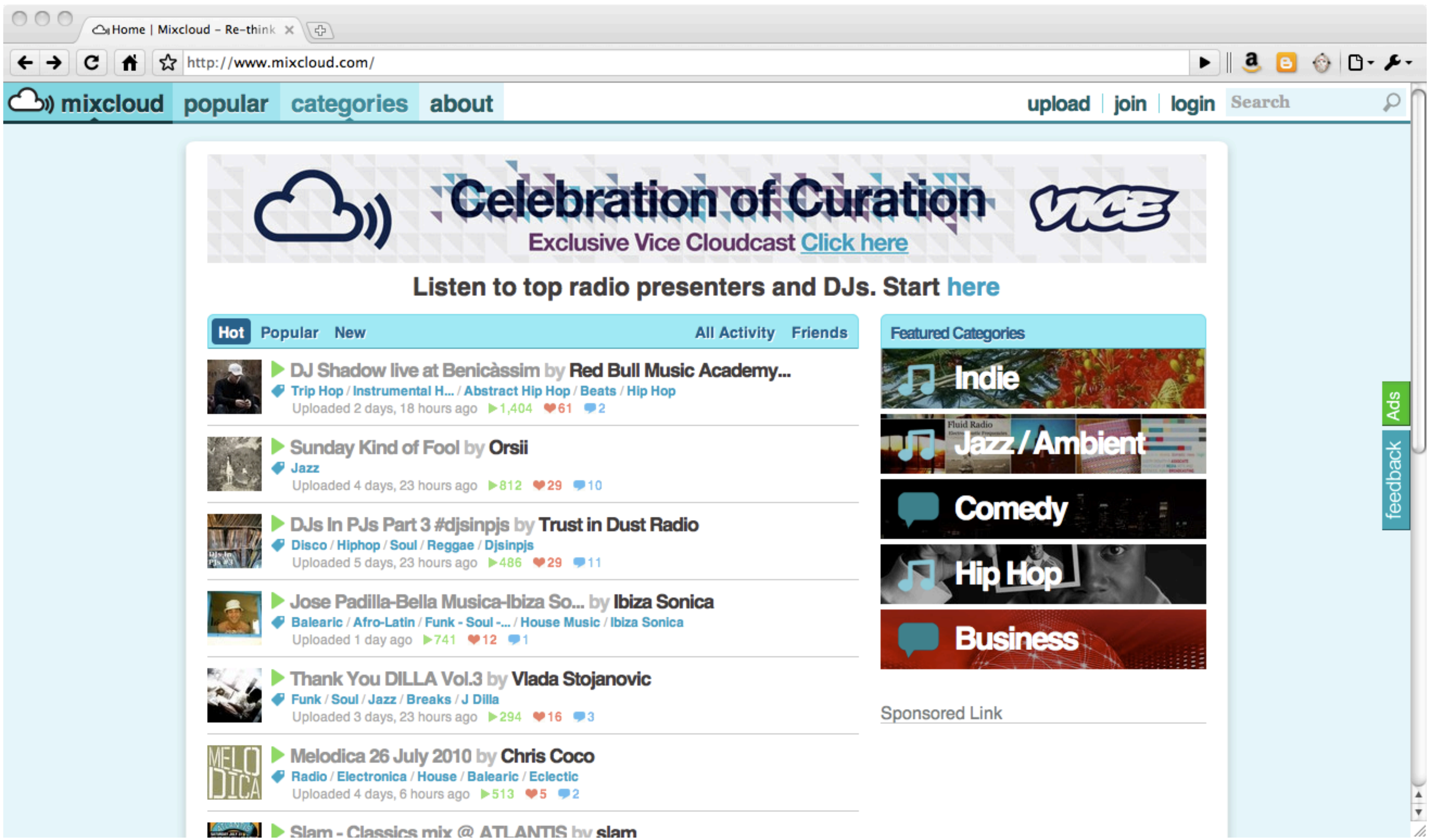
Spotify

- Sharable playlists
- Collaborative playlists
- Many 3rd party playlist sites



Mix Enablers

mixcloud

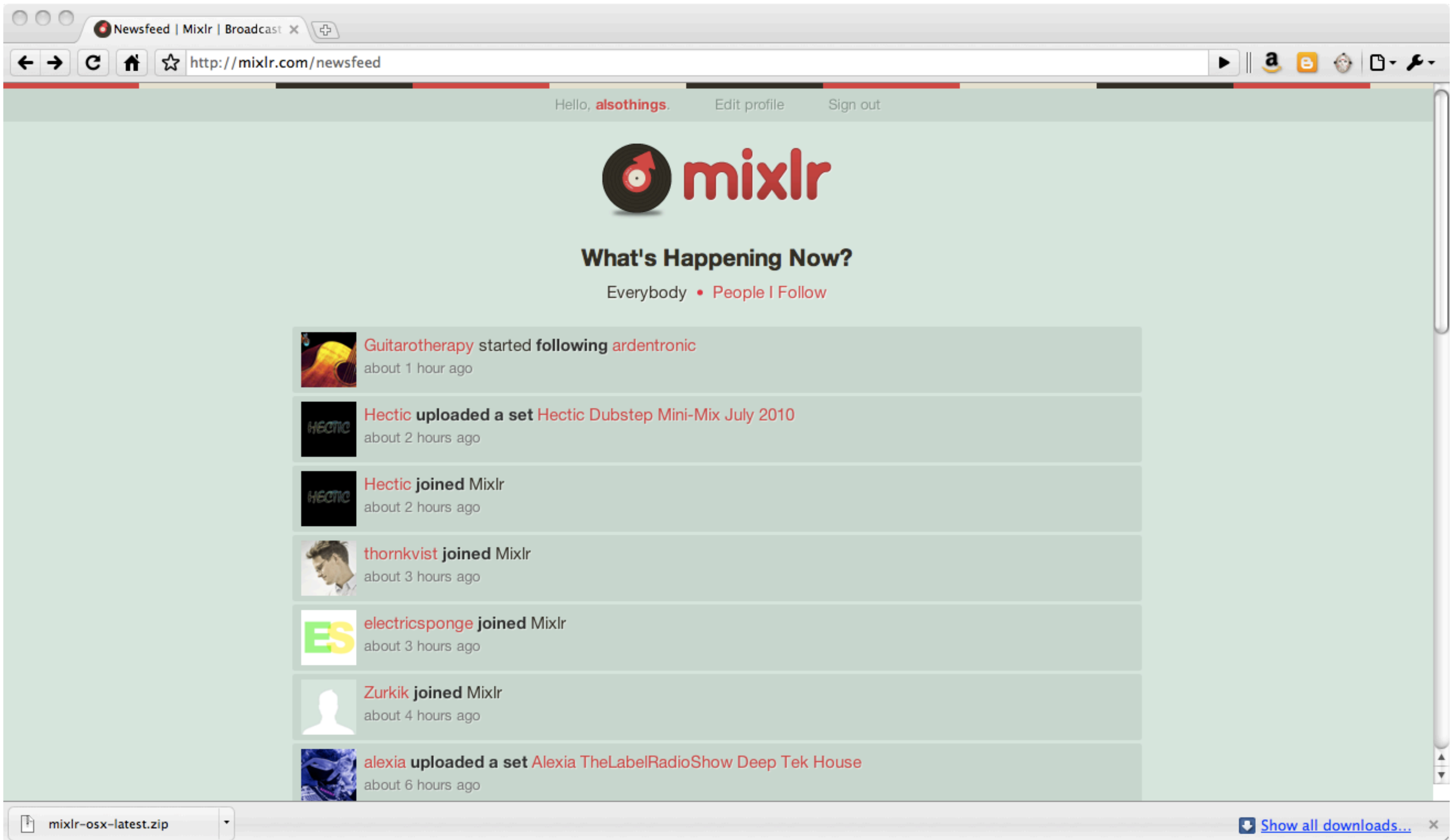


Mix Enablers
mixcloud

- Free social networking platform organized around the exchange of long form audio, principally [dance] music
- Provides a means for DJs (aspiring and professional) to connect with the audience and into the Web of Things

Mix Enablers

mixlr



Mix Enablers
mixlr

- Focused on adding social features to centralized multicasting
- Supports live and recorded (mixed and unmixed) streams
- Social connectivity is web-based, broadcaster is a native application
- Native app provides integration with common DJ tools

setlist.fm

A wiki for concert setlists

setlist.fm

A wiki for concert setlists

They have an API!

Emerson, Lake & Palmer Concert at Victoria Park, London, England Setlist on July 25, 2010

Jul 25

Artist
Emerson, Lake & Palmer   Artist statistics   Add setlist
Venue
Victoria Park, London, England
Attendees
aeolist cafcchegs Steban14
Last edited
July 31, 2010 2:04:17 PM UTC by Blackadder

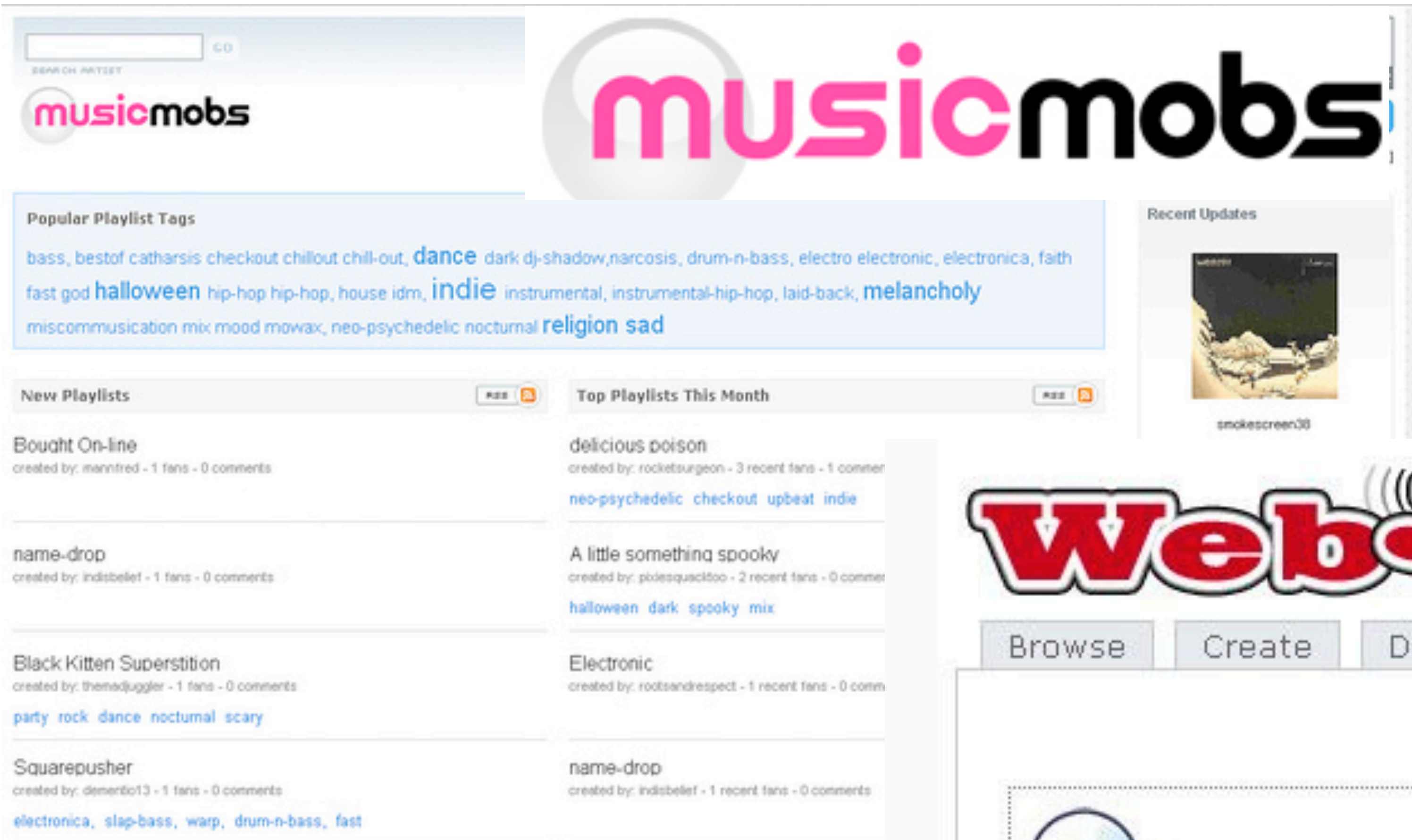
SHARE

1. *Karn Evil 9: 1st Impression, Part 2*
2. *The Barbarian*
3. *Bitches Crystal*
4. *Touch and Go*
5. *Knife-Edge*
6. *From The Beginning*
7. *Take a Pebble*
8. *Tarkus*
9. *Farewell to Arms*
10. *Lucky Man*
11. *Pictures at an Exhibition*
12. *Fanfare for the Common Man/Drum Solo/Rondo*

REST Endpoints

- /0.1/artist/{mbid}
- /0.1/city/{geoId}
- /0.1/search/artists
- /0.1/search/cities
- /0.1/search/countries
- /0.1/search/setlists
- /0.1/search/venues
- /0.1/setlist/{setlistId}
- /0.1/venue/{venueId}
- /0.1/artist/{mbid}/setlists
- /0.1/setlist/lastFm/{lastFmEventId}
- /0.1/setlist/version/{versionId}
- /0.1/venue/{venueId}/setlists
- /0.1/artist/{mbid}/tour/{tour}

The playlisting dead pool



research systems

Human-Facilitating Systems

Personal Radio

- An early collaborative filtering system
- Users rated songs directly
- Playlists are built by finding similar (via Pearson's correlation coefficient) users
- Playlists can, once built, be streamed, named, shared and **modified**
- Order is either random or user defined

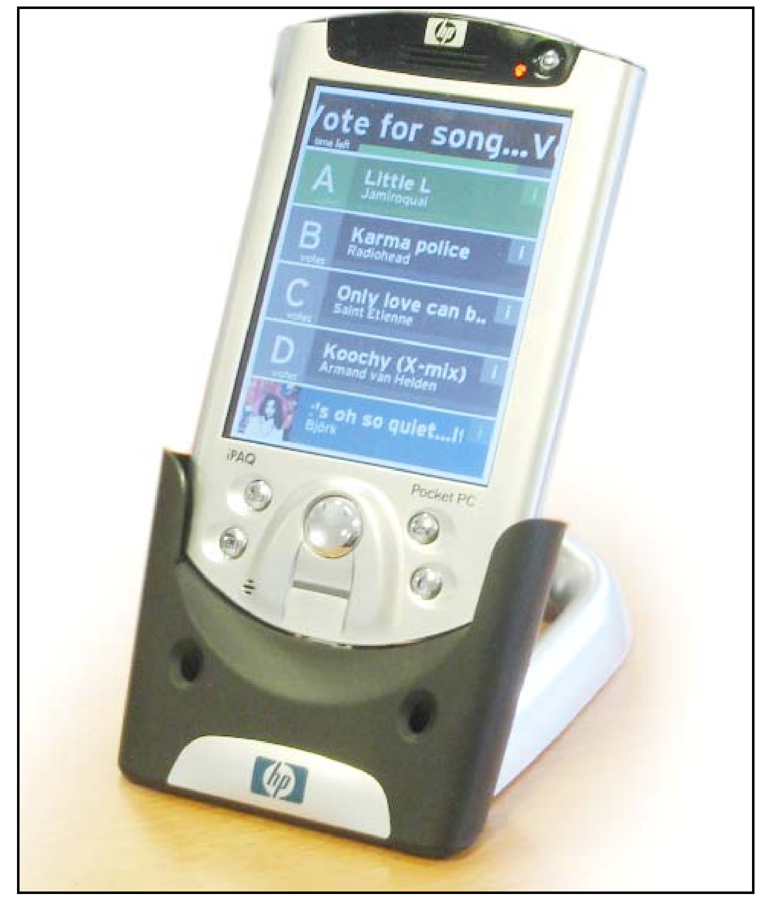
Personal Radio

- An early collaborative filtering system
- Users rated songs directly
- Playlists are built by finding similar (via Pearson's correlation coefficient) users
- Playlists can, once built, be streamed, named, shared and **modified**
- Order is either random or user defined

Smart radio: Building music radio on the fly
Conor Hayes and Pádraig Cunningham

Collaborative Choice

A public voting system



Jukola: democratic music choice in a public space
K. O'Hara, M. Lipson, M. Jansen, A. Unger, H. Jeffries, and P. Macer

Collaborative Choice

Decentralized supply

Jukola: democratic music choice in a public space
K. O'Hara, M. Lipson, M. Jansen, A. Unger, H. Jeffries, and P. Macer

Playlist Sharing

- Music should help convey status information and implicit presence
- Music should help build interpersonal relationships
- A good individual listening experience should be supported
- Support smooth continuous use

Playlist Sharing

1. Members associate music from their personal library to their activities and locations

2. For each new song, the system picks a random user and a song from that user's current state

3. Music is streamed to each mobile device

4. The device displays the current song and which user assigned it

Social playlist: enabling touch points and enriching ongoing relationships through collaborative mobile music listening
KuanTing Liu and Roger Andersson Reimer

Field Tested:

- Music should help convey status information and implicit presence
- Music should help build interpersonal relationships
- A good individual listening experience should be supported
- Support smooth continuous use

Field Tested:

- Music should help convey status information and implicit presence
- Music should help build interpersonal relationships
- A good individual listening experience should be supported
- Support smooth continuous use

"I am a weather guy. Happy music for sunny days so to speak."

Field Tested:

- Music should help convey status information and implicit presence
- Music should help build interpersonal relationships
- A good individual listening experience should be supported
- Support smooth continuous use

"I am a weather guy. Happy music for sunny days so to speak."

"I made her a CD because I can't stand her music."

Field Tested:

- Music should help convey status information and implicit presence
- Music should help build interpersonal relationships
- A good individual listening experience should be supported
- Support smooth continuous use

"I am a weather guy. Happy music for sunny days so to speak."

“I made her a CD because I can’t stand her music.”

Participants report on hearing between 30% - 50% “bad songs”.

Field Tested:

- Music should help convey status information and implicit presence

"I am a weather guy. Happy music for sunny days so to speak."

- Music should help build interpersonal relationships

“I made her a CD because I can’t stand her music.”

- A good individual listening experience should be supported

Participants report on hearing between 30% - 50% “bad songs”.

- Support smooth continuous use

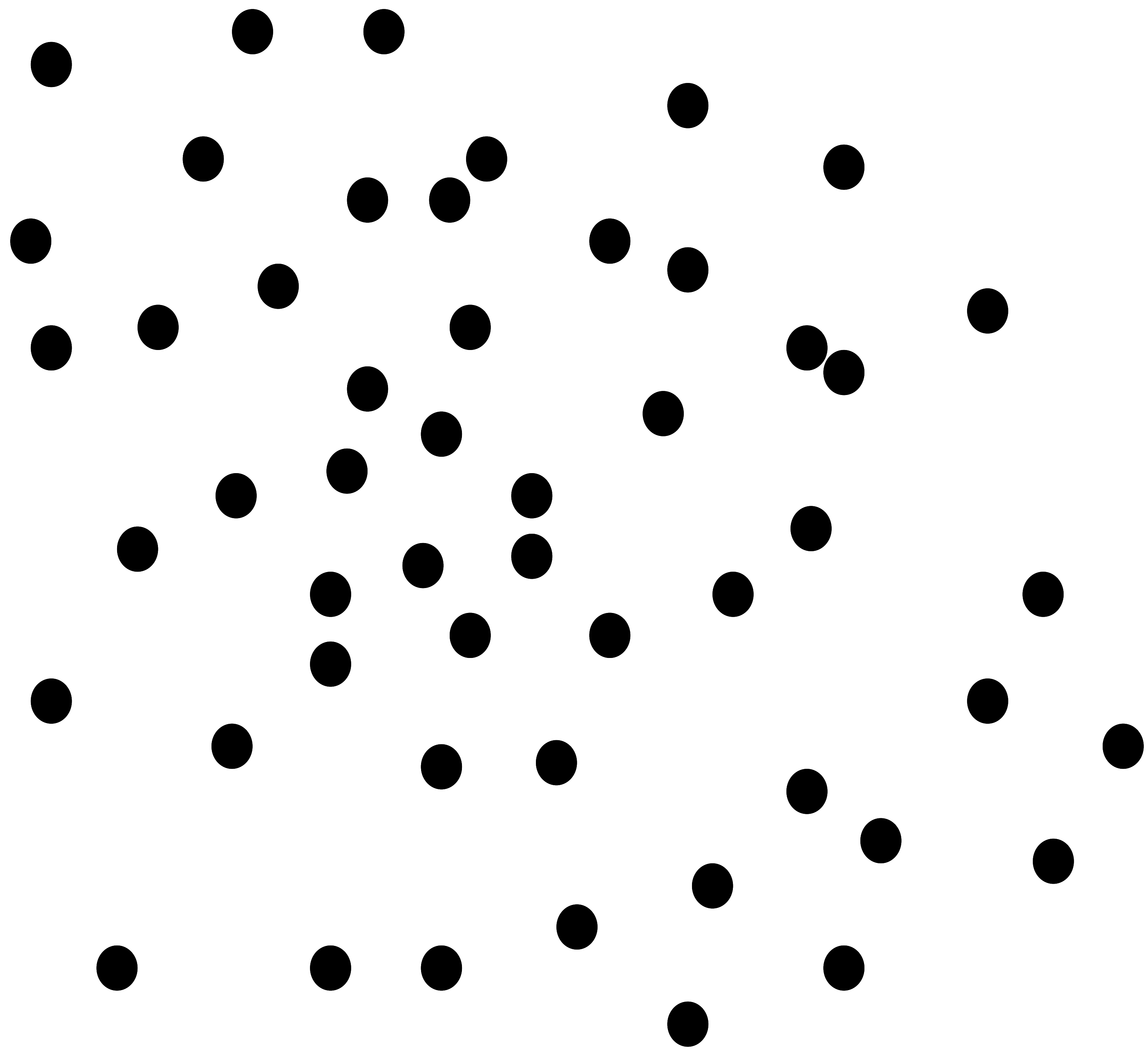
At such occasions, they may turn off the service and switch to their own music library.

Implications

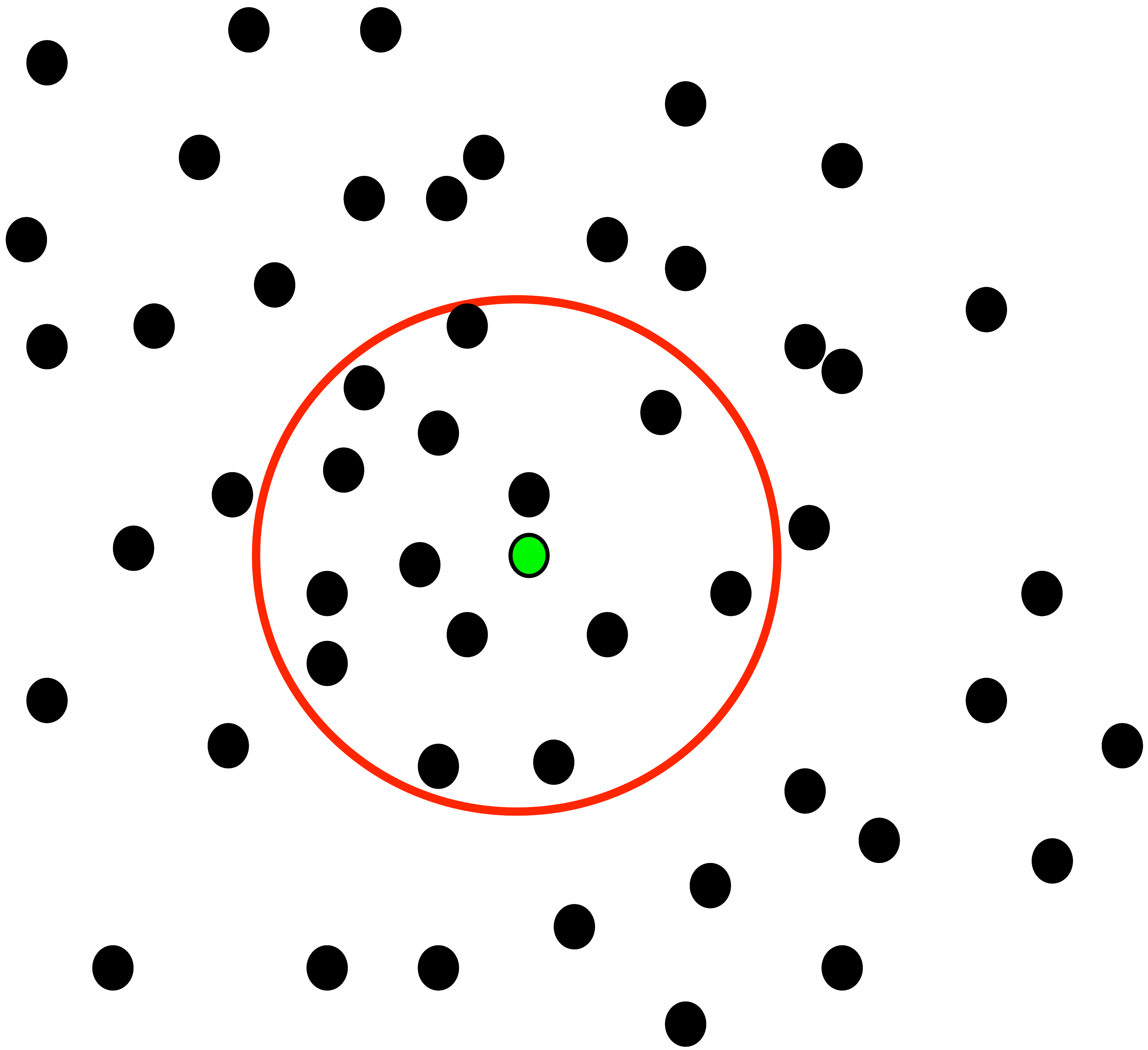
- Smooth integration with individual music listening to encourage continuous use
- Allow flexibility and cues to support self-expression and enable touch points
- Support ongoing relationships
- Counterbalance experiences of bad songs and misinterpretations

Fully Automatic Systems

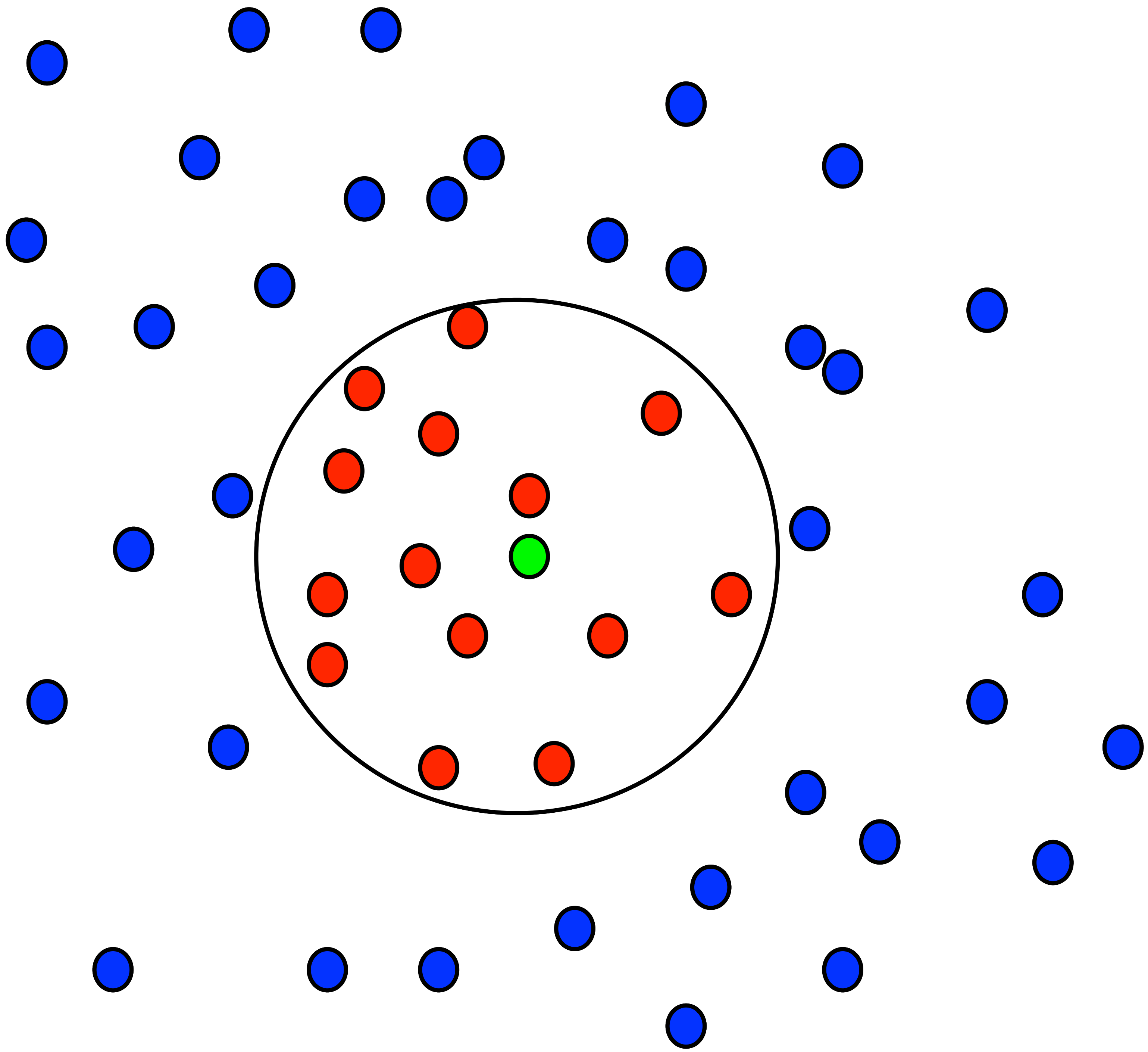
Nearest Neighbors



Nearest Neighbors



Nearest Neighbors



Pure Content

- Uses MFCCs and finds *N* nearest neighbors
- Forms a graph with all songs weighted by distance
- Playlist is created by finding the shortest weighted path covering *N* songs

Pure Content

Relevance	Average nr. of relevant songs in playlist		
	Size 5	Size 10	Size 20
Same Genre	3.46	6.60	12.6
Same Artist	1.34	2.07	3.01
Same Album	1.11	1.63	2.21

Relevance	Scheme	Average nr. of relevant songs in playlist		
		Size 5	Size 10	Size 20
Same Genre	Trajectory,1	3.26	6.13	10.75
Same Artist		1.08	1.43	1.68
Same Album		0.89	1.11	1.22
Same Genre	Trajectory,2	3.33	6.37	12.08
Same Artist		1.23	1.89	2.73
Same Album		1.01	1.49	2.00
Same Genre	Feedback	3.40	6.54	12.46
Same Artist		1.27	1.96	2.83
Same Album		1.05	1.54	2.07

Content-Based Playlist Generation: Exploratory Experiments
Beth Logan

Pure Content

Relevance	Average nr. of relevant songs in playlist		
	Size 5	Size 10	Size 20
Same Genre	3.46	6.60	12.6
Same Artist	1.34	2.07	3.01
Same Album	1.11	1.63	2.21

Relevance	Scheme	Average nr. of relevant songs in playlist		
		Size 5	Size 10	Size 20
Same Genre	Trajectory,1	3.26	6.13	10.75
Same Artist		1.08	1.43	1.68
Same Album		0.89	1.11	1.22
Same Genre	Trajectory,2	3.33	6.37	12.08
Same Artist		1.23	1.89	2.73
Same Album		1.01	1.49	2.00
Same Genre	Feedback	3.40	6.54	12.46
Same Artist		1.27	1.96	2.83
Same Album		1.05	1.54	2.07

Content-Based Playlist Generation: Exploratory Experiments
Beth Logan

Pure Content

Relevance	Average nr. of relevant songs in playlist		
	Size 5	Size 10	Size 20
Same Genre	3.46	6.60	12.6
Same Artist	1.34	2.07	3.01
Same Album	1.11	1.63	2.21

Relevance	Scheme	Average nr. of relevant songs in playlist		
		Size 5	Size 10	Size 20
Same Genre	Trajectory,1	3.26	6.13	10.75
Same Artist		1.08	1.43	1.68
Same Album		0.89	1.11	1.22
Same Genre	Trajectory,2	3.33	6.37	12.08
Same Artist		1.23	1.89	2.73
Same Album		1.01	1.49	2.00
Same Genre	Feedback	3.40	6.54	12.46
Same Artist		1.27	1.96	2.83
Same Album		1.05	1.54	2.07

Content-Based Playlist Generation: Exploratory Experiments
Beth Logan

Metadata Models

Metadata Field	Example Values	Number of Values
Genre	Jazz, Reggae, Hip-Hop	30
Subgenre	Heavy Metal, I'm So Sad and Spaced Out	572
Style	East Coast Rap, Gangsta Rap, West Coast Rap	890
Mood	Dreamy, Fun, Angry	21
Rhythm Type	Straight, Swing, Disco	10
Rhythm Description	Frenetic, Funky, Lazy	13
Vocal Code	Instrumental, Male, Female, Duet	6

Metadata Models

Metadata Field	Example Values	Number of Values
Genre	Jazz, Reggae, Hip-Hop	30
Subgenre	Heavy Metal, I'm So Sad and Spaced Out	572
Style	East Coast Rap, Gangsta Rap, West Coast Rap	890
Mood	Dreamy, Fun, Angry	21
Rhythm Type	Straight, Swing, Disco	10
Rhythm Description	Frenetic, Funky, Lazy	13
Vocal Code	Instrumental, Male, Female, Duet	6

- Use Gaussian Process Regression to create playlists based on seed tracks

Learning a Gaussian Process Prior for Automatically Generating Music Playlists
John C. Platt and Christopher J.C. Burges and Steven Swenson and Christopher Weare and Alice Zheng

Metadata Models

Metadata Field	Example Values	Number of Values
Genre	Jazz, Reggae, Hip-Hop	30
Subgenre	Heavy Metal, I'm So Sad and Spaced Out	572
Style	East Coast Rap, Gangsta Rap, West Coast Rap	890
Mood	Dreamy, Fun, Angry	21
Rhythm Type	Straight, Swing, Disco	10
Rhythm Description	Frenetic, Funky, Lazy	13
Vocal Code	Instrumental, Male, Female, Duet	6

- Use Gaussian Process Regression to create playlists based on seed tracks
- Using Kernel Meta-Training algorithm on albums to select the priors

Learning a Gaussian Process Prior for Automatically Generating Music Playlists
John C. Platt and Christopher J.C. Burges and Steven Swenson and Christopher Weare and Alice Zheng

Metadata Models

	Playlist 1	Playlist 2
Seed	Eagles, The Sad Cafe	Eagles, Life in the Fast Lane
1	Genesis, More Fool Me	Eagles, Victim of Love
2	Bee Gees, Rest Your Love On Me	Rolling Stones, Ruby Tuesday
3	Chicago, If You Leave Me Now	Led Zeppelin, Communication Breakdown
4	Eagles, After The Thrill Is Gone	Creedence Clearwater, Sweet Hitch-hiker
5	Cat Stevens, Wild World	Beatles, Revolution

- Use Gaussian Process Regression to create playlists based on seed tracks
- Using Kernel Meta-Training algorithm on albums to select the priors
- Playlists are formed based on the maximum log likelihood from the selected seed song

Learning a Gaussian Process Prior for Automatically Generating Music Playlists
John C. Platt and Christopher J.C. Burges and Steven Swenson and Christopher Weare and Alice Zheng

Metadata Models

Playlist Method	Number of Seed Songs 1	2	3	4	5	6	7	8	9
KMT + GPR	**42.9**	**46.0**	**44.8**	43.8	**46.8**	**45.0**	**44.2**	**44.4**	**44.8**
Hamming + GPR	32.7	39.2	39.8	39.6	41.3	40.0	39.5	38.4	39.8
Hamming + No GPR	32.7	39.0	39.6	40.2	42.6	41.4	41.5	41.7	43.2
Random Order	6.3	6.6	6.5	6.2	6.5	6.6	6.2	6.1	6.8

- Use Gaussian Process Regression to create playlists based on seed tracks
- Using Kernel Meta-Training algorithm on albums to select the priors
- Playlists are formed based on the maximum log likelihood from the selected seed song

Learning a Gaussian Process Prior for Automatically Generating Music Playlists
John C. Platt and Christopher J.C. Burges and Steven Swenson and Christopher Weare and Alice Zheng

Metadata Models

Playlist Method	Number of Seed Songs 1	2	3	4	5	6	7	8	9
KMT + GPR	**42.9**	**46.0**	**44.8**	43.8	**46.8**	**45.0**	**44.2**	**44.4**	**44.8**
Hamming + GPR	32.7	39.2	39.8	39.6	41.3	40.0	39.5	38.4	39.8
Hamming + No GPR	32.7	39.0	39.6	40.2	42.6	41.4	41.5	41.7	43.2
Random Order	6.3	6.6	6.5	6.2	6.5	6.6	6.2	6.1	6.8

- Use Gaussian Process Regression to create playlists based on seed tracks
- Using Kernel Meta-Training algorithm on albums to select the priors
- Playlists are formed based on the maximum log likelihood from the selected seed song

Learning a Gaussian Process Prior for Automatically Generating Music Playlists
John C. Platt and Christopher J.C. Burges and Steven Swenson and Christopher Weare and Alice Zheng

Metadata Models

Playlist Method	Number of Seed Songs								
	1	2	3	4	5	6	7	8	9
KMT + GPR	**42.9**	**46.0**	**44.8**	43.8	**46.8**	**45.0**	**44.2**	**44.4**	**44.8**
Hamming + GPR	32.7	39.2	39.8	39.6	41.3	40.0	39.5	38.4	39.8
Hamming + No GPR	32.7	39.0	39.6	40.2	42.6	41.4	41.5	41.7	43.2
Random Order	6.3	6.6	6.5	6.2	6.5	6.6	6.2	6.1	6.8

- Use Gaussian Process Regression to create playlists based on seed tracks
- Using Kernel Meta-Training algorithm on albums to select the priors
- Playlists are formed based on the maximum log likelihood from the selected seed song

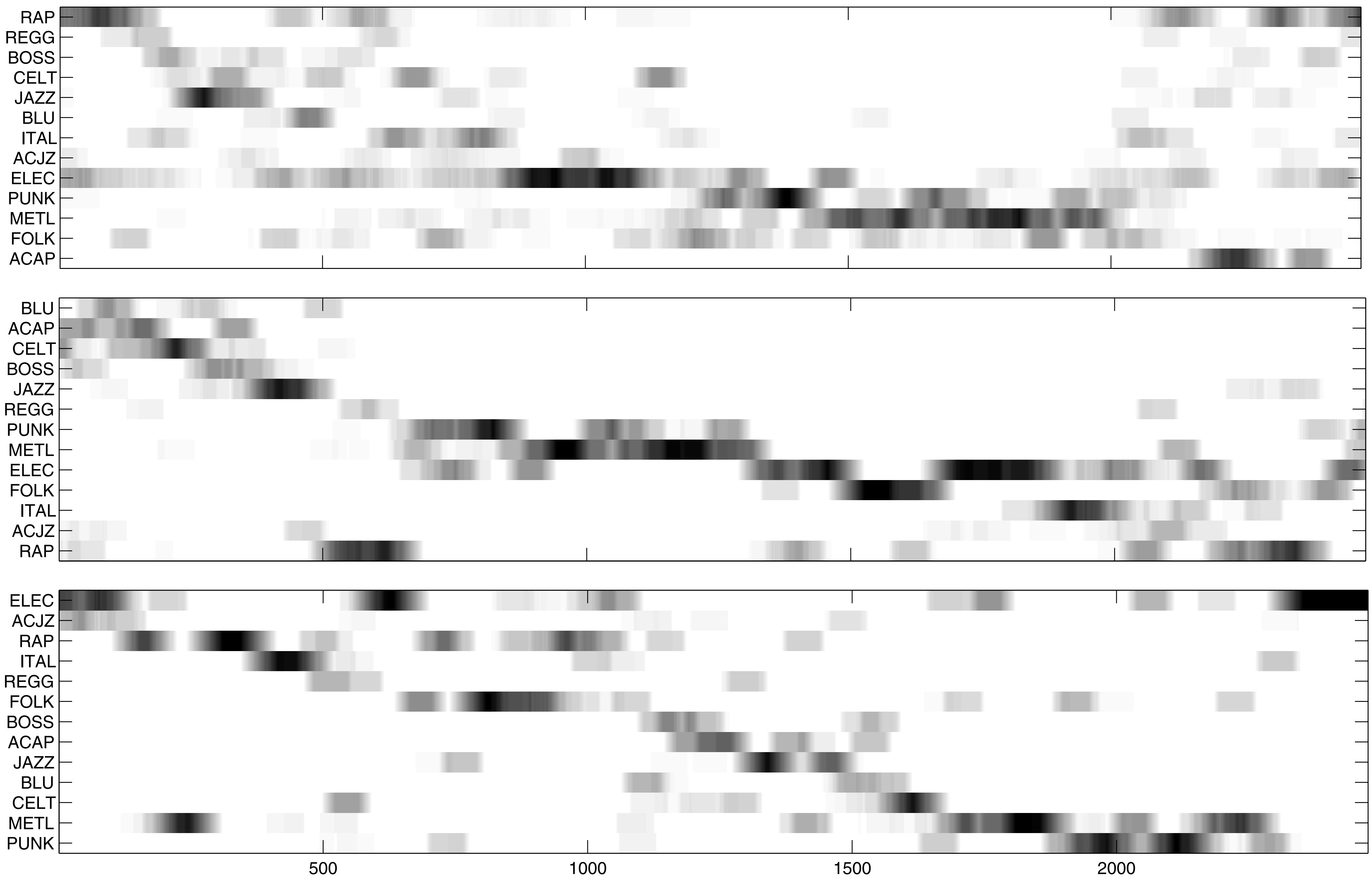
Learning a Gaussian Process Prior for Automatically Generating Music Playlists
John C. Platt and Christopher J.C. Burges and Steven Swenson and Christopher Weare and Alice Zheng

Traveling Sales Playlist?

Traveling Sales Playlist?

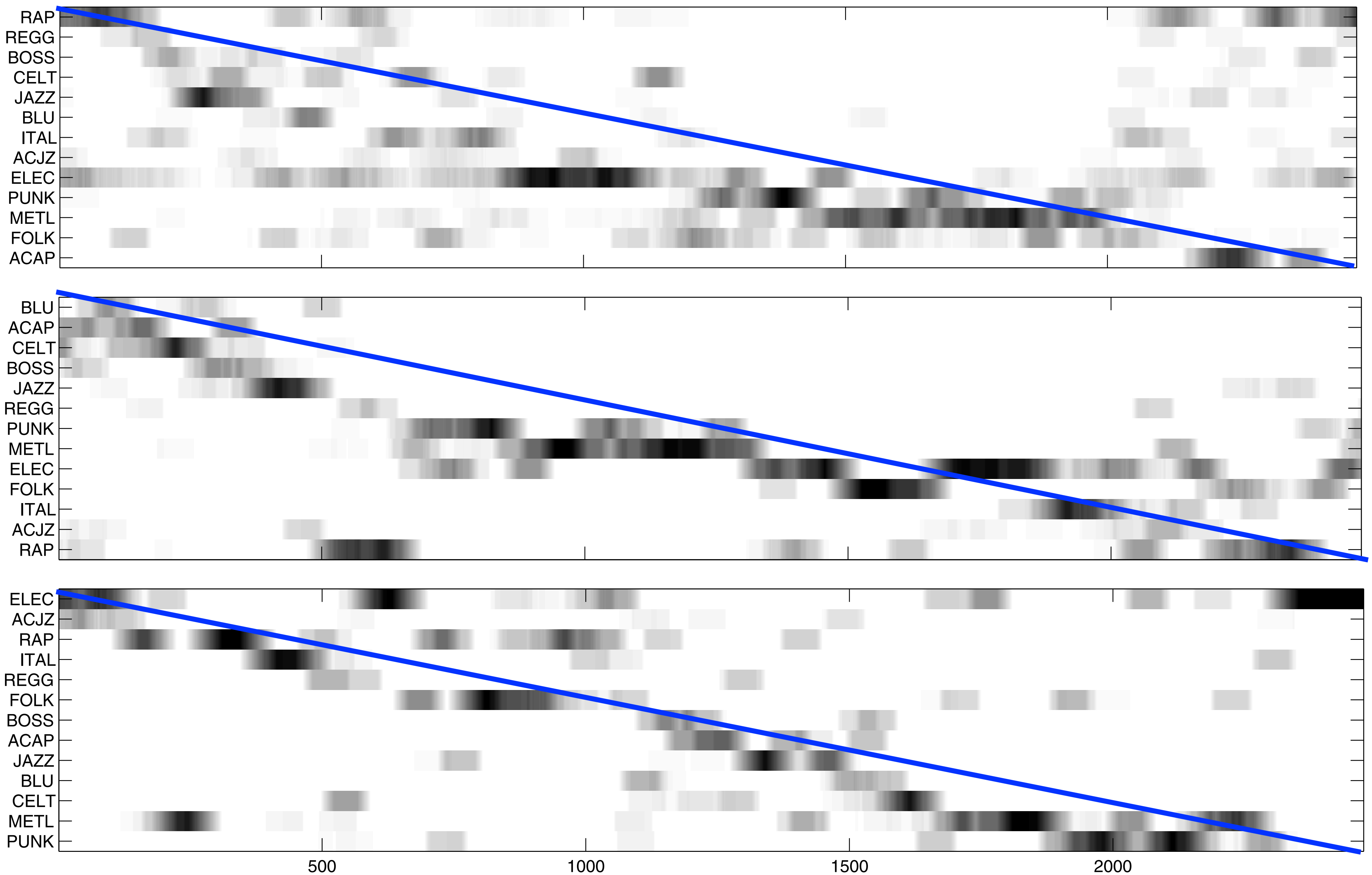
- Using a combination of content-based song and web-based artist similarity to generate a distance matrix
- Approximation of TSP is used to find 'tours' through the collection
- Tested on two collections of about 3000 tracks

Now With Web Data



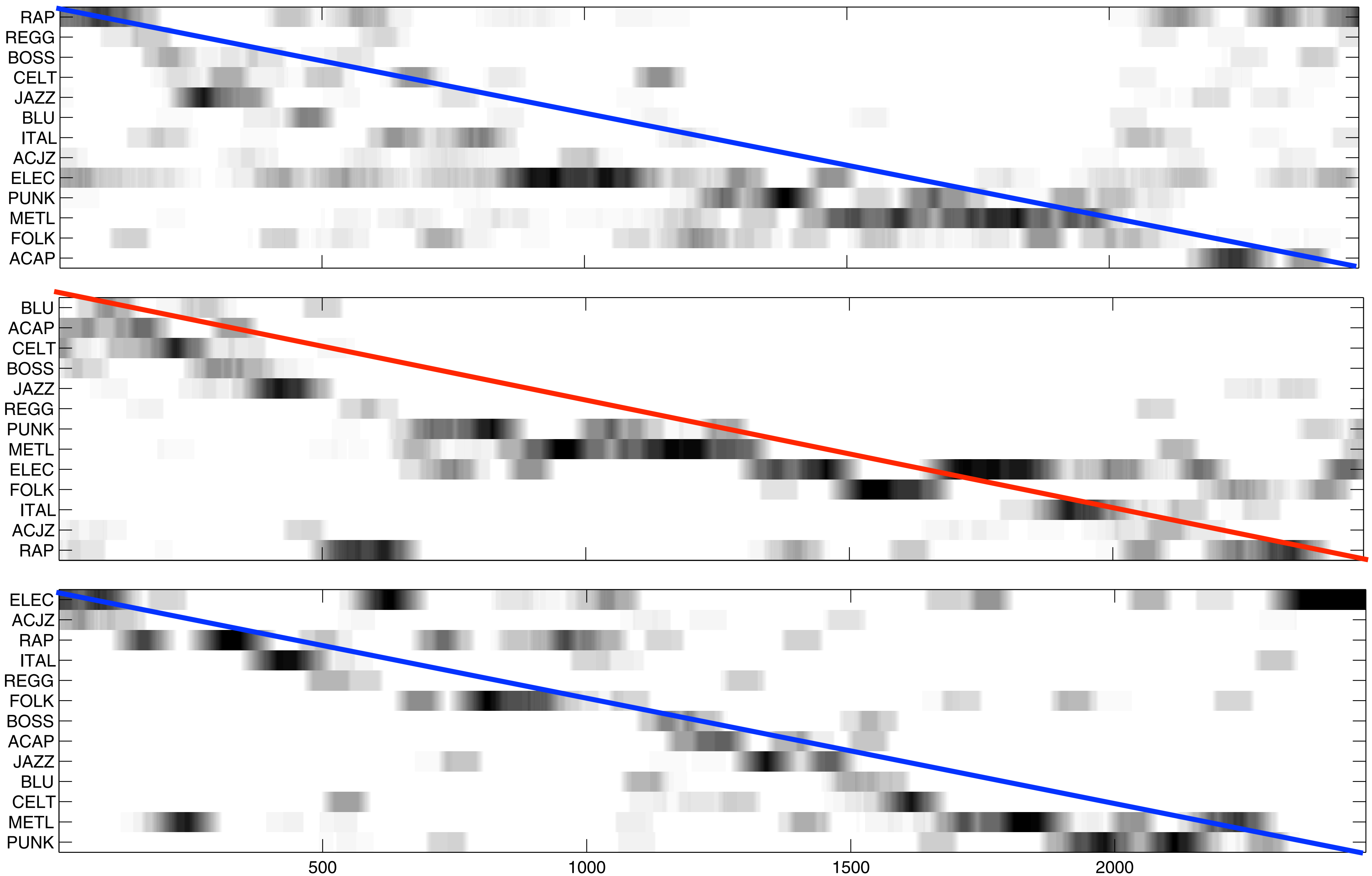
Combining Audio-based Similarity with Web-based Data to Accelerate Automatic Music Playlist Generation
Peter Knees, Tim Pohle, Markus Schedl, and Gerhard Widmer

Now With Web Data



Combining Audio-based Similarity with Web-based Data to Accelerate Automatic Music Playlist Generation
Peter Knees, Tim Pohle, Markus Schedl, and Gerhard Widmer

Now With Web Data



Combining Audio-based Similarity with Web-based Data to Accelerate Automatic Music Playlist Generation
Peter Knees, Tim Pohle, Markus Schedl, and Gerhard Widmer

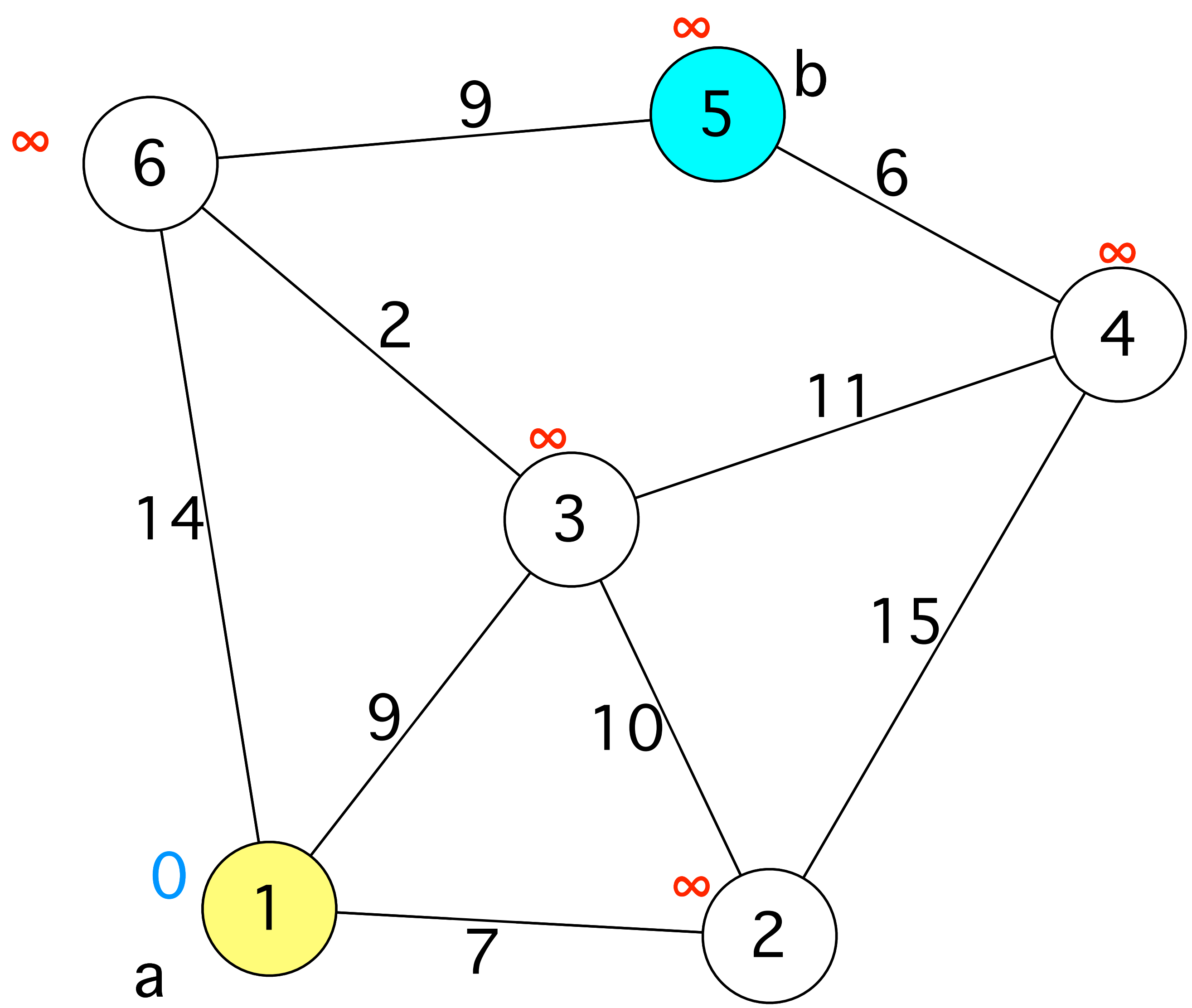
Graph Methods

Dijkstra's algorithm

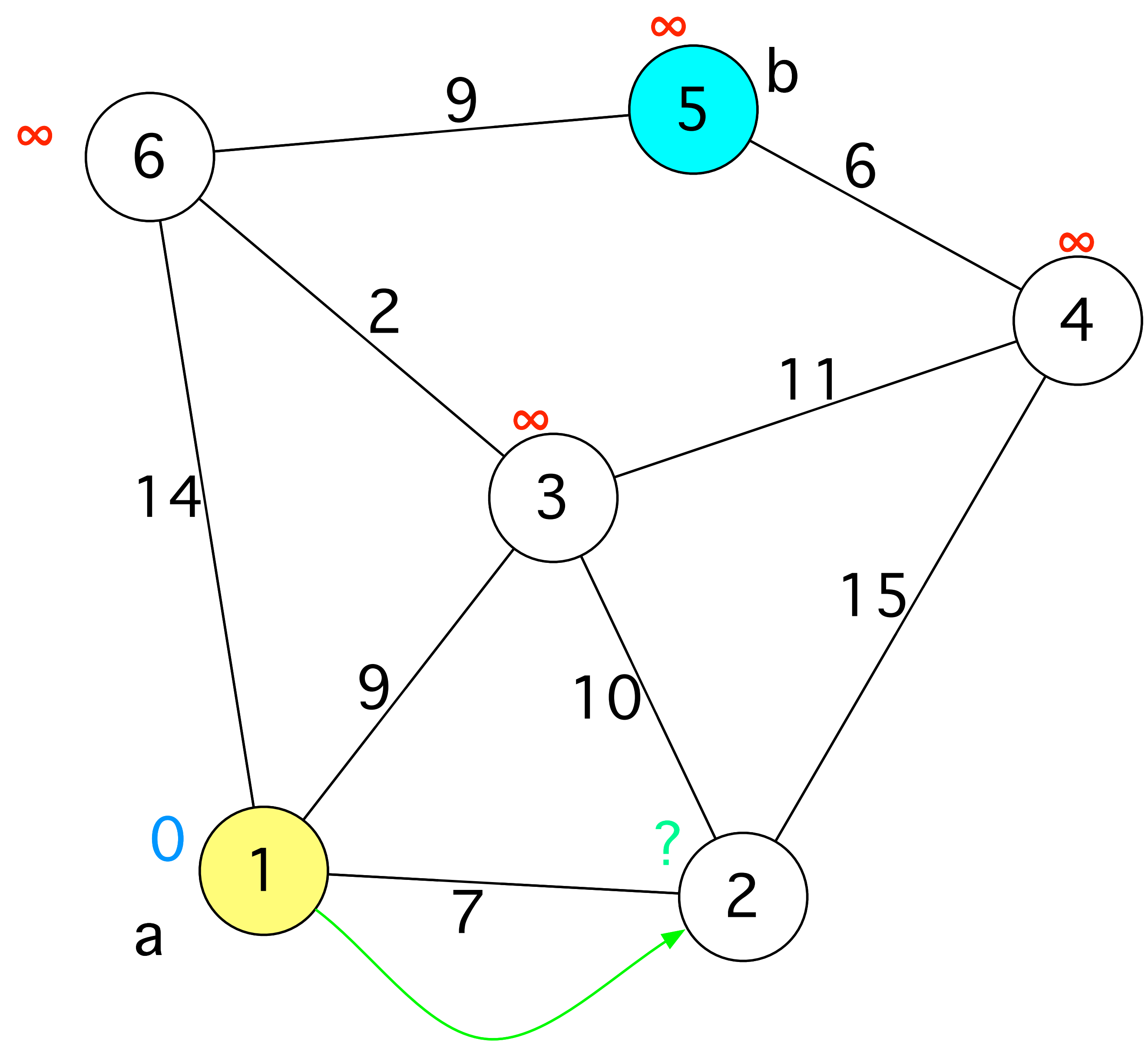
1. Assign to every node a distance value. Set it to zero for our initial node and to infinity for all other nodes.
2. Mark all nodes as unvisited. Set initial node as current.
3. For current node, consider all its unvisited neighbors and calculate their tentative distance (from the initial node).
4. When we are done considering all neighbors of the current node, mark it as visited. A visited node will not be checked ever again; its distance recorded now is final and minimal.
5. If all nodes have been visited, finish. Otherwise, set the unvisited node with the smallest distance (from the initial node) as the next "current node" and continue from step 3.

Graph Methods

Dijkstra's algorithm

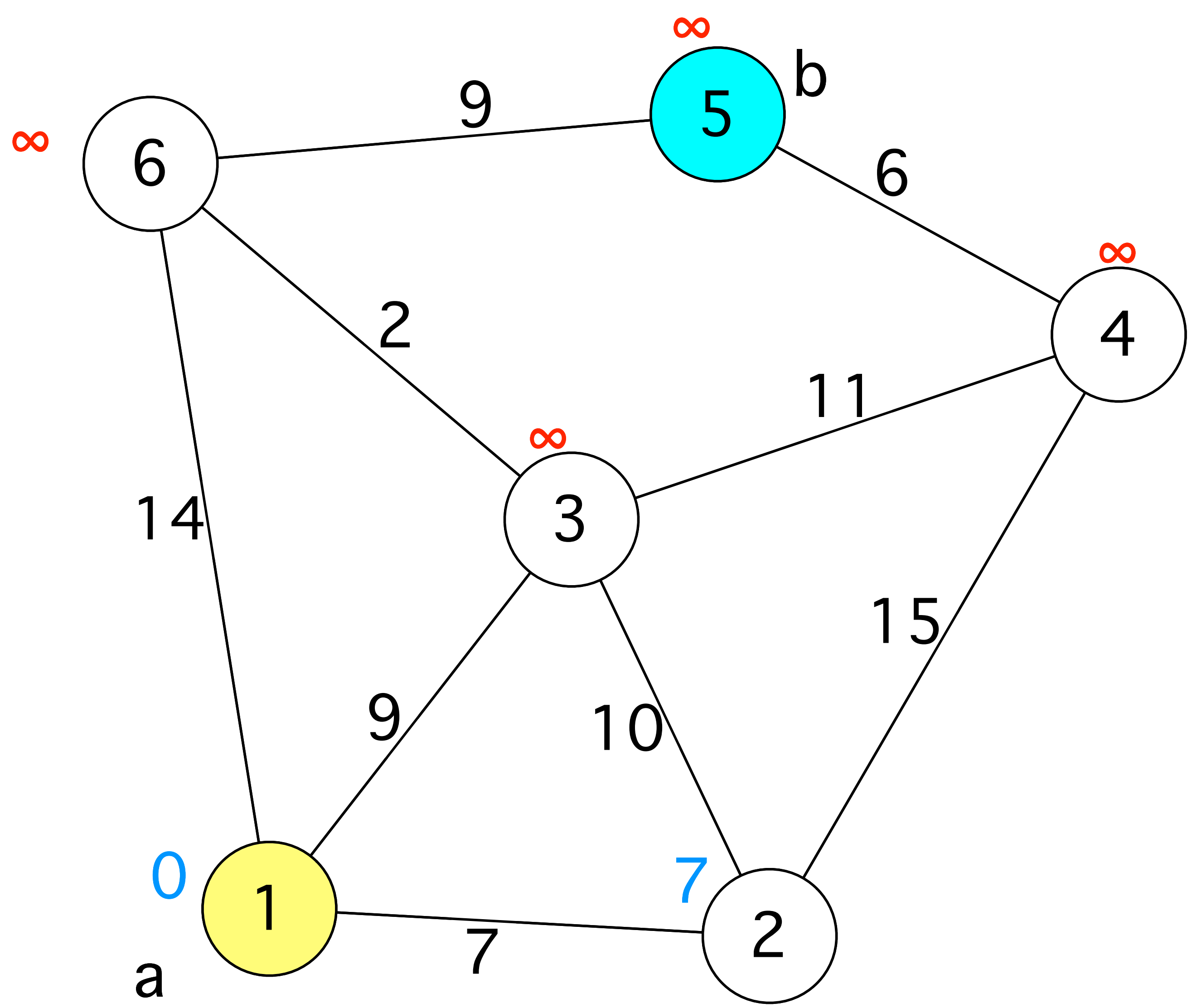


Graph Methods
Dijkstra's algorithm



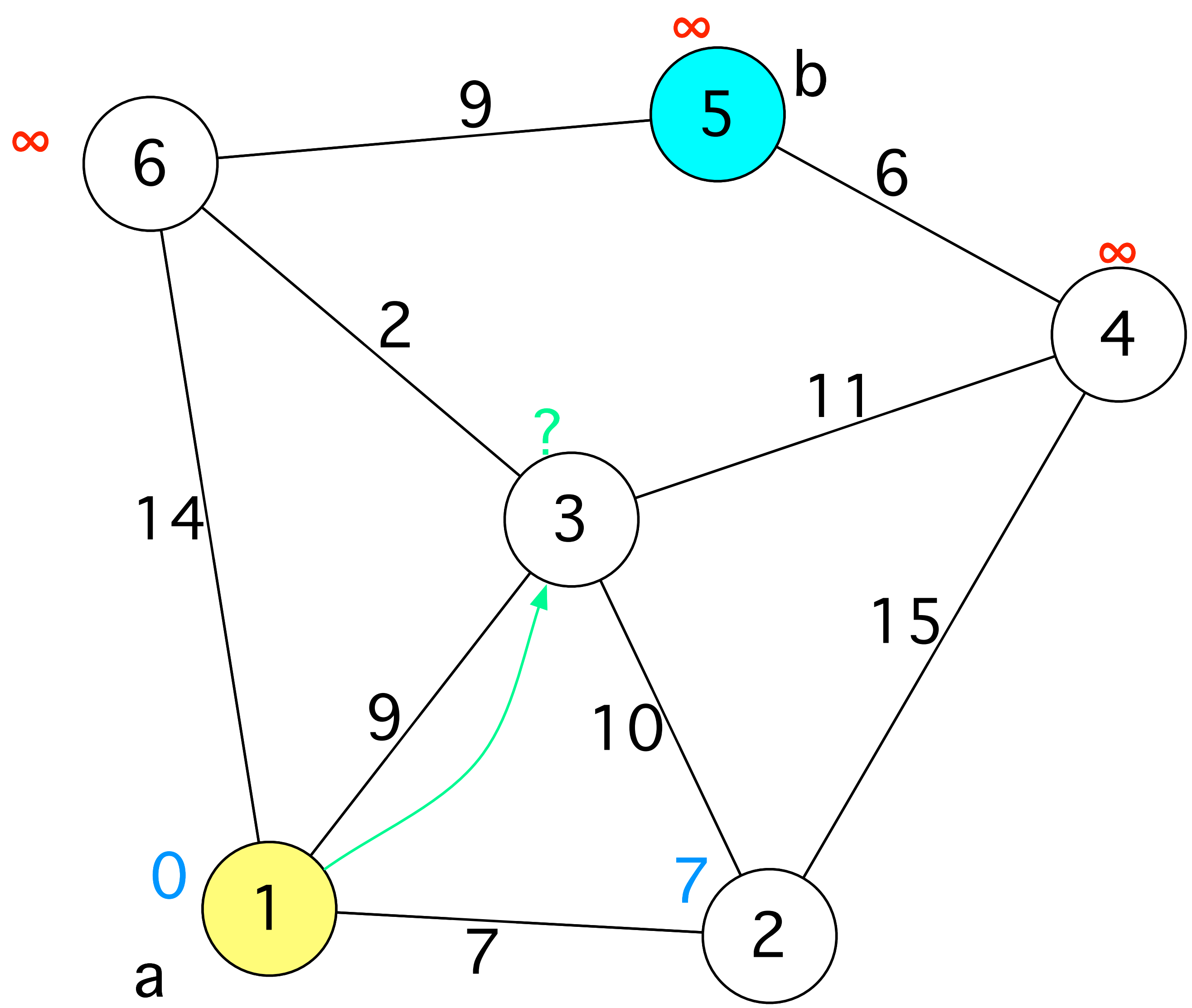
Graph Methods

Dijkstra's algorithm



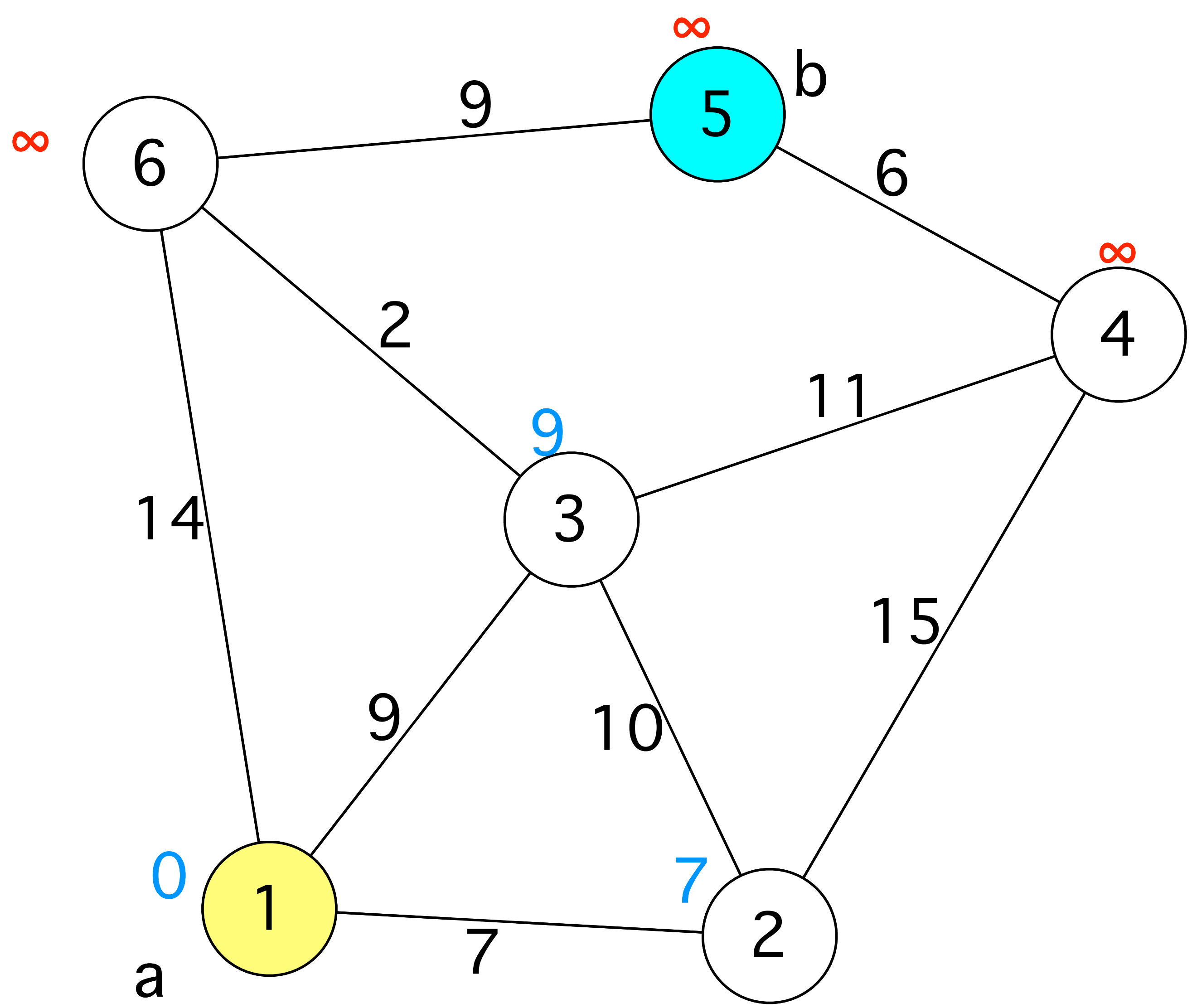
Graph Methods

Dijkstra's algorithm



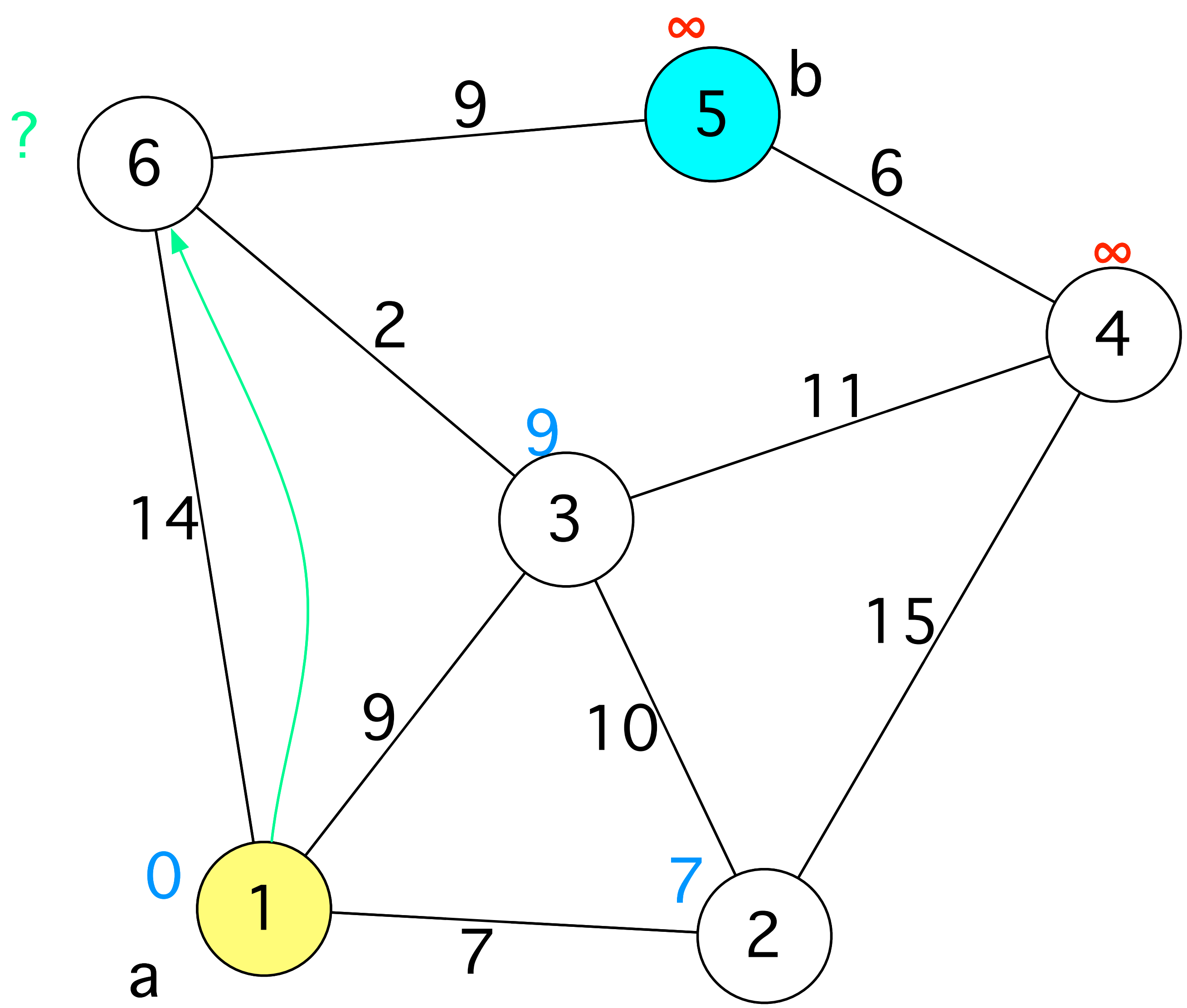
Graph Methods

Dijkstra's algorithm



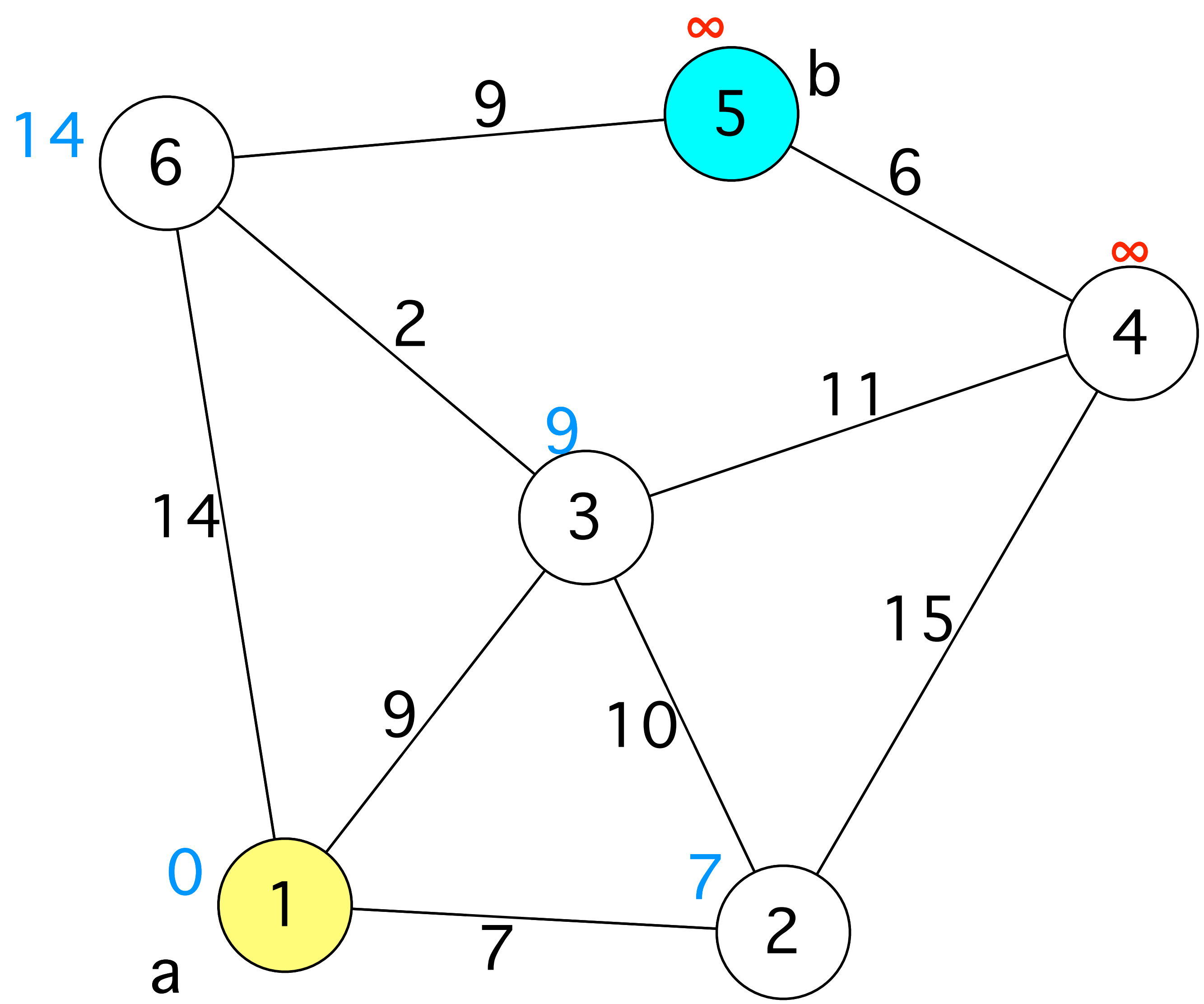
Graph Methods

Dijkstra's algorithm



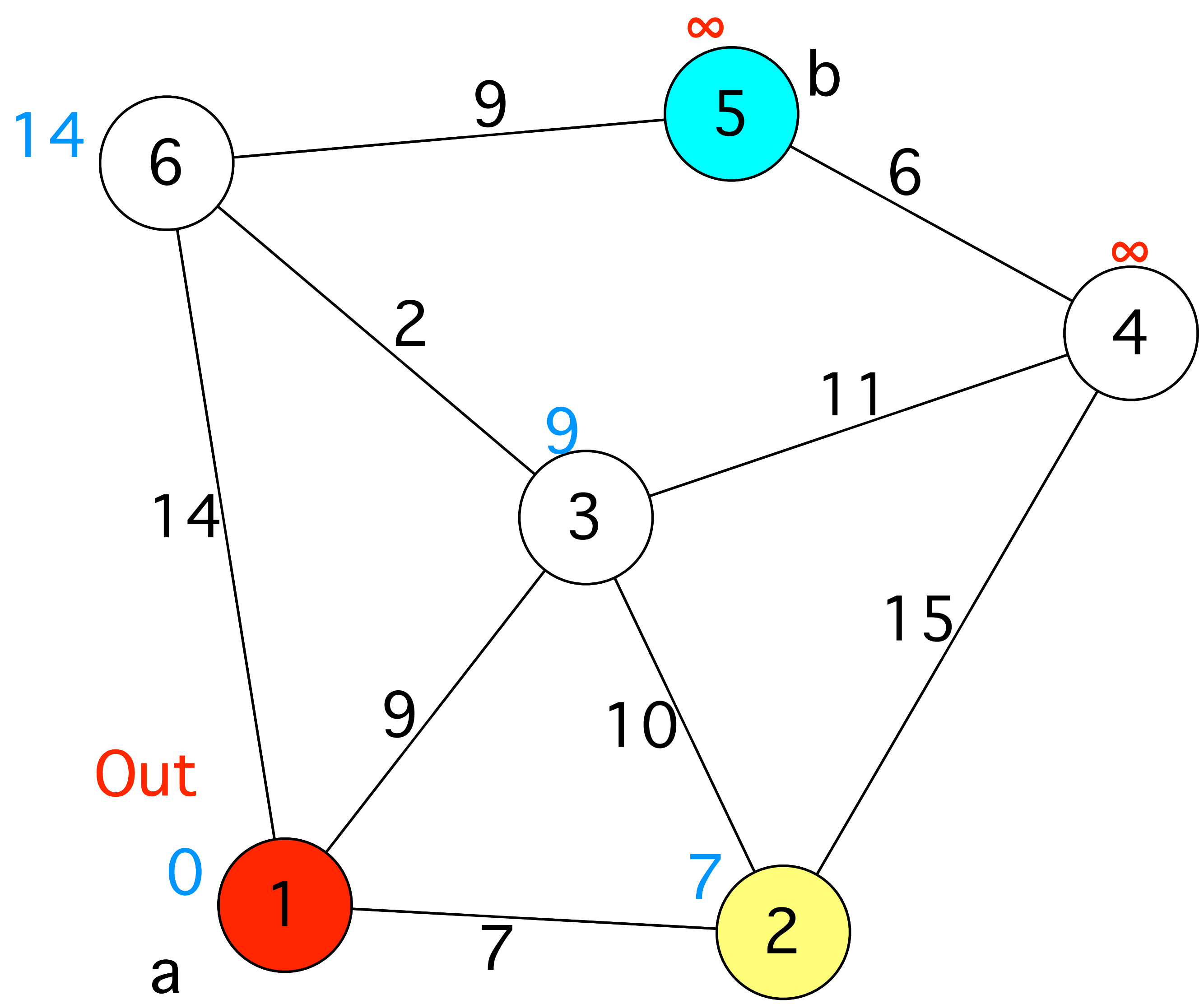
Graph Methods

Dijkstra's algorithm



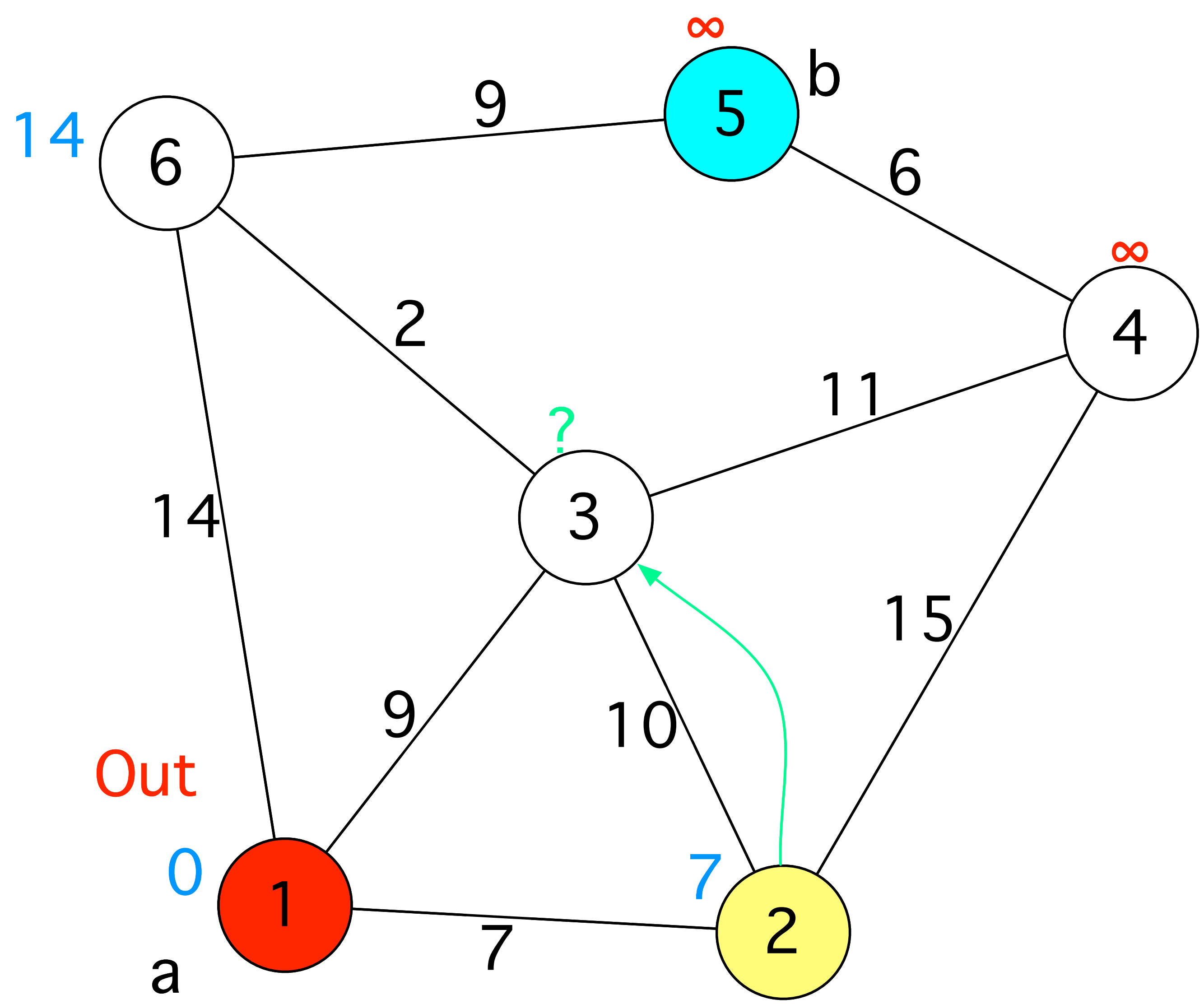
Graph Methods

Dijkstra's algorithm



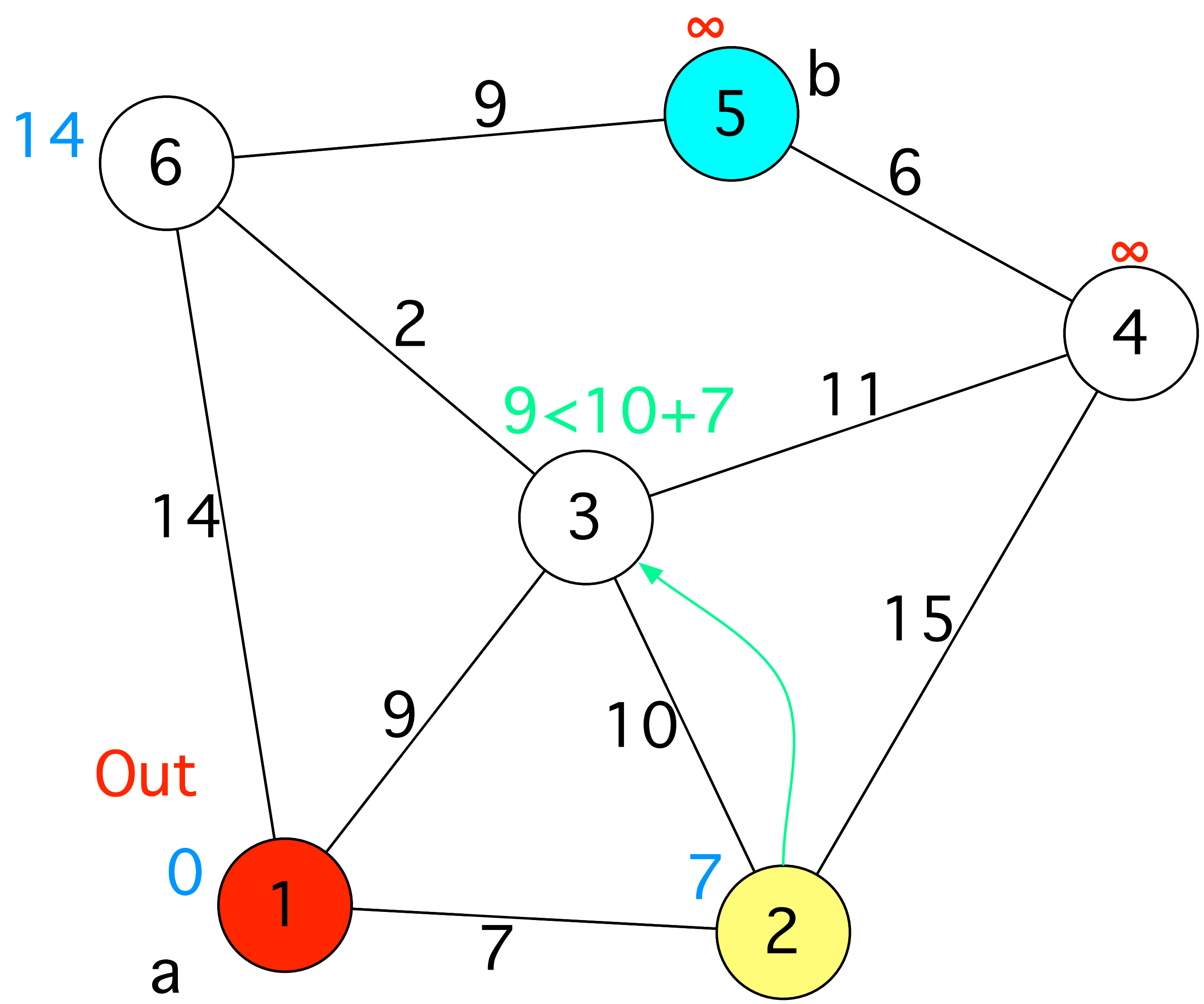
Graph Methods

Dijkstra's algorithm



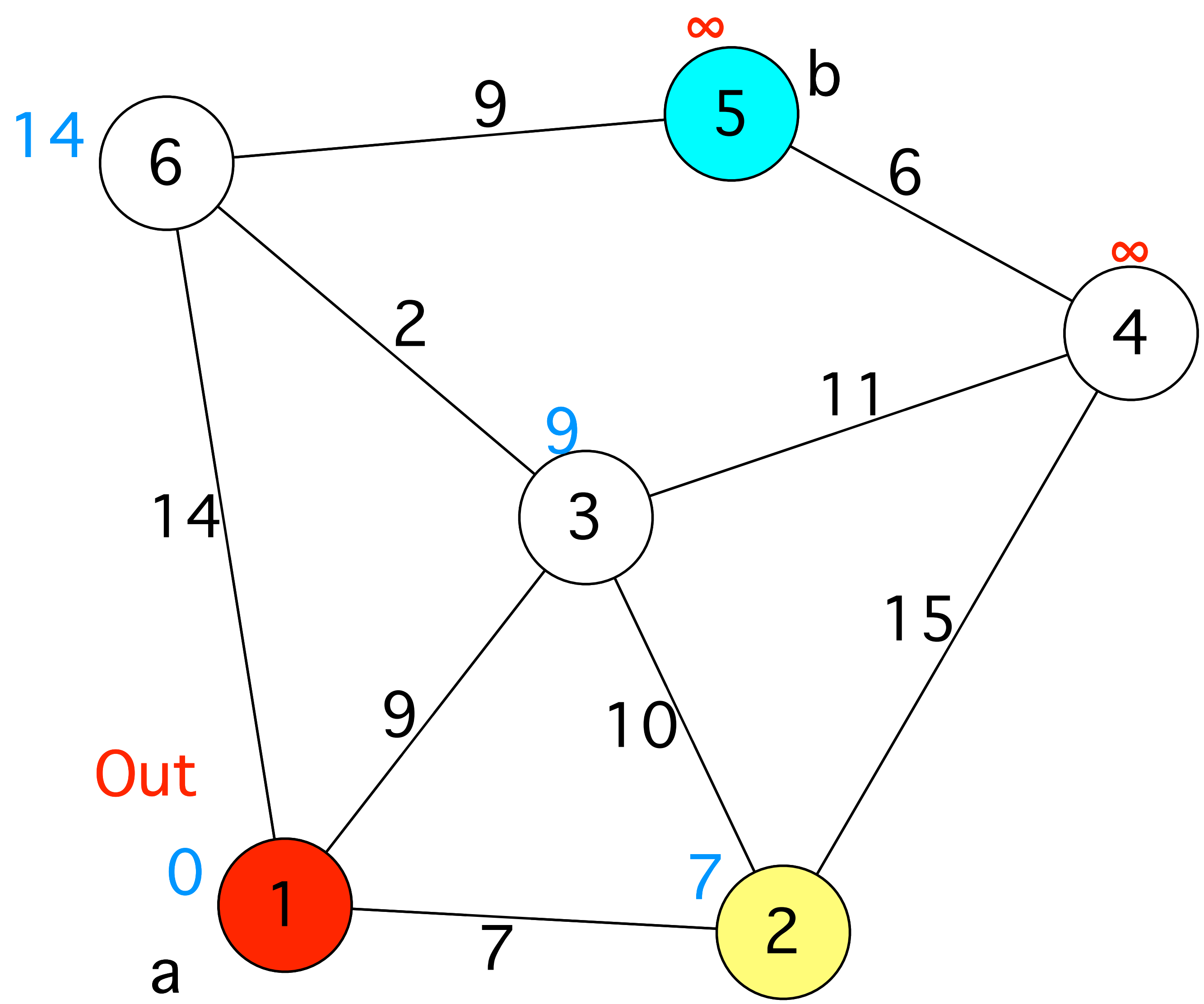
Graph Methods

Dijkstra's algorithm



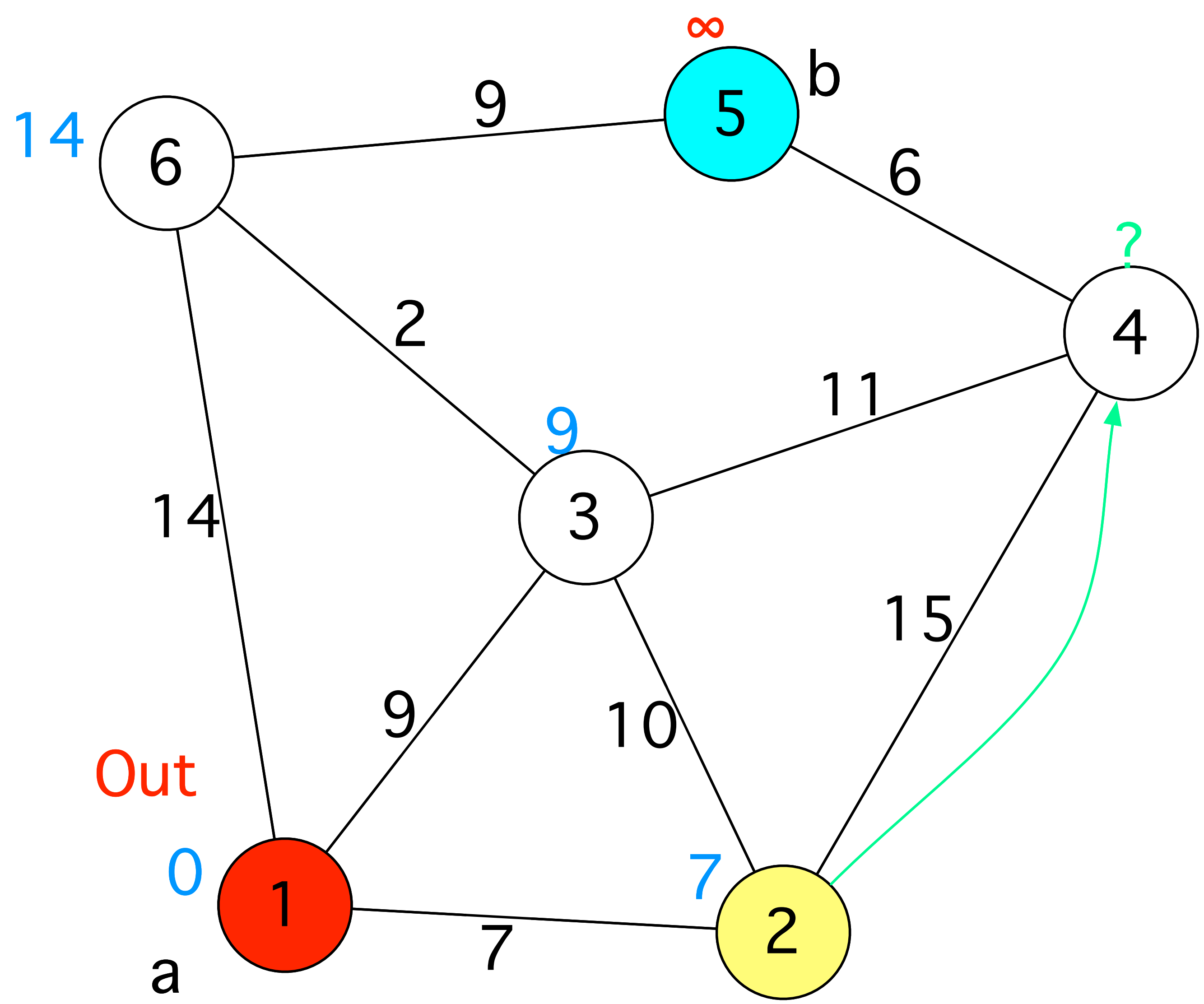
Graph Methods

Dijkstra's algorithm



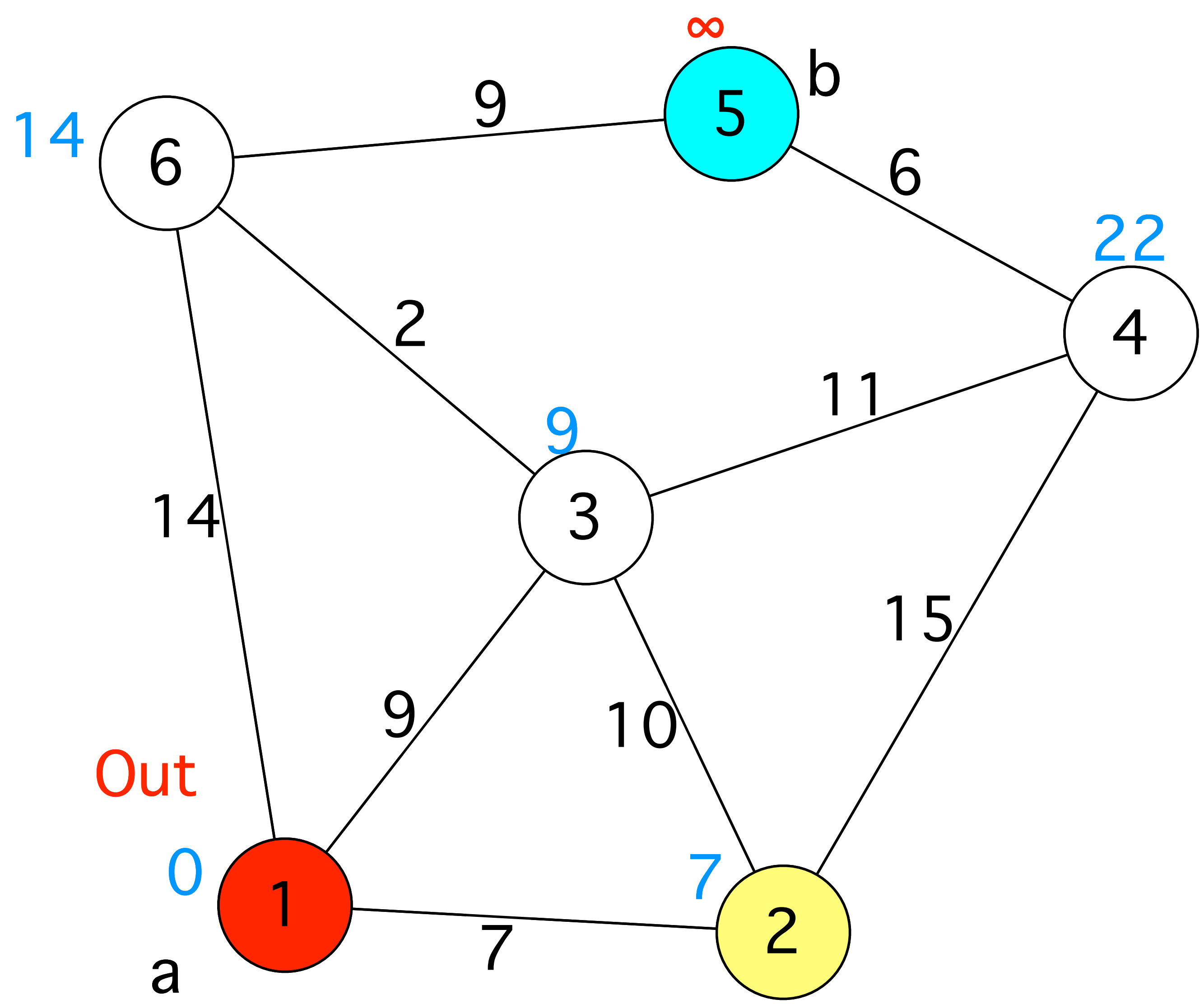
Graph Methods

Dijkstra's algorithm



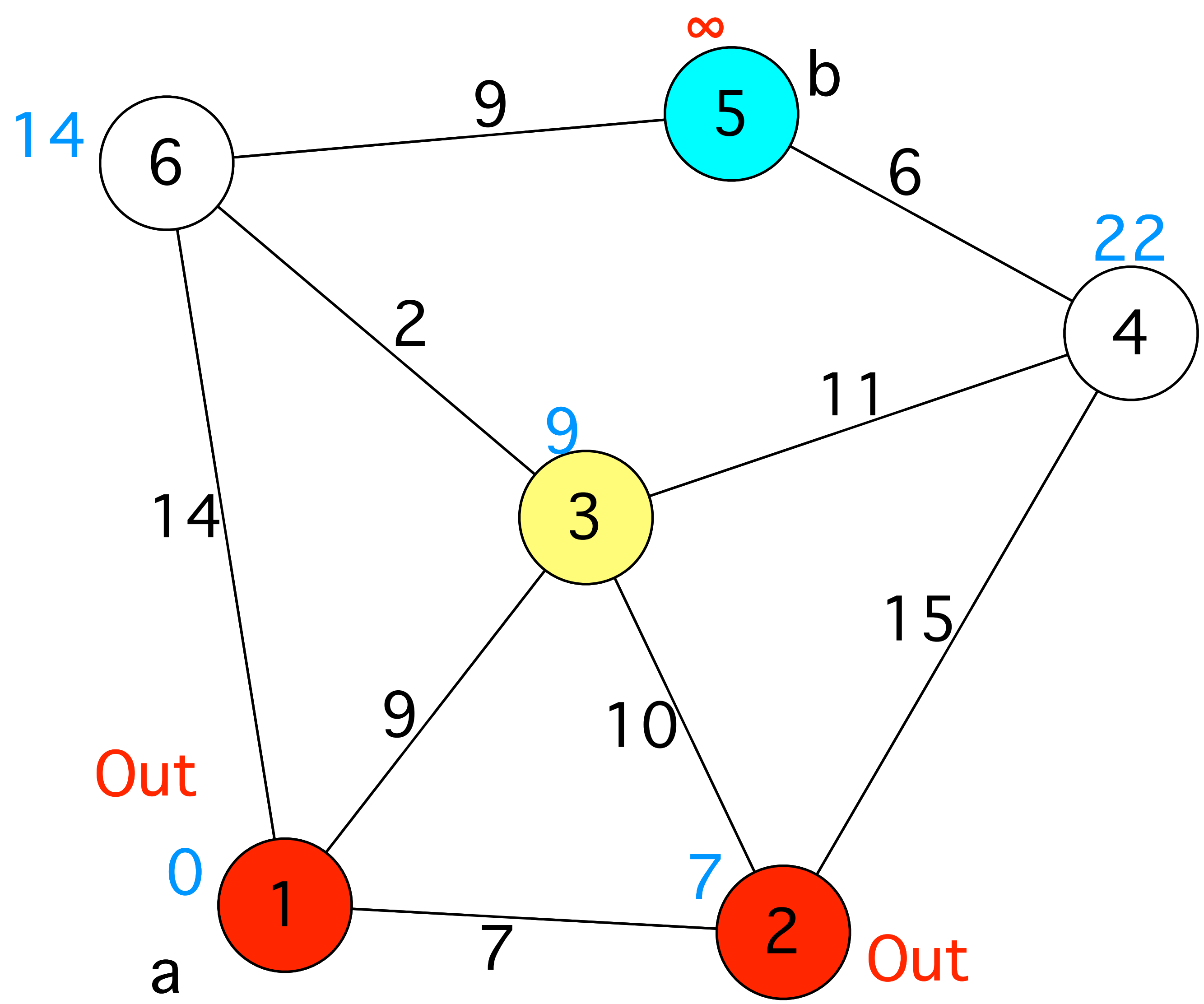
Graph Methods

Dijkstra's algorithm



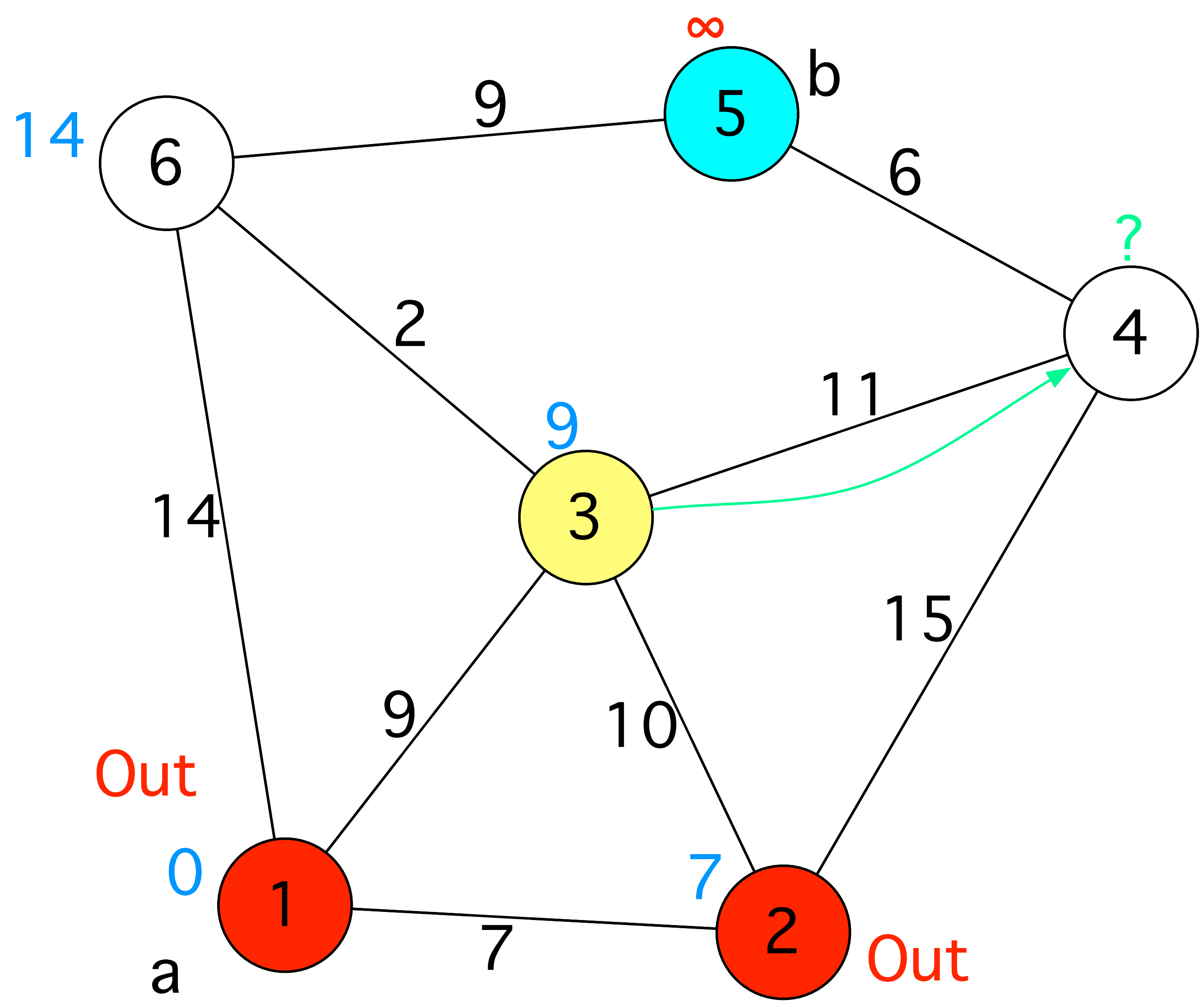
Graph Methods

Dijkstra's algorithm



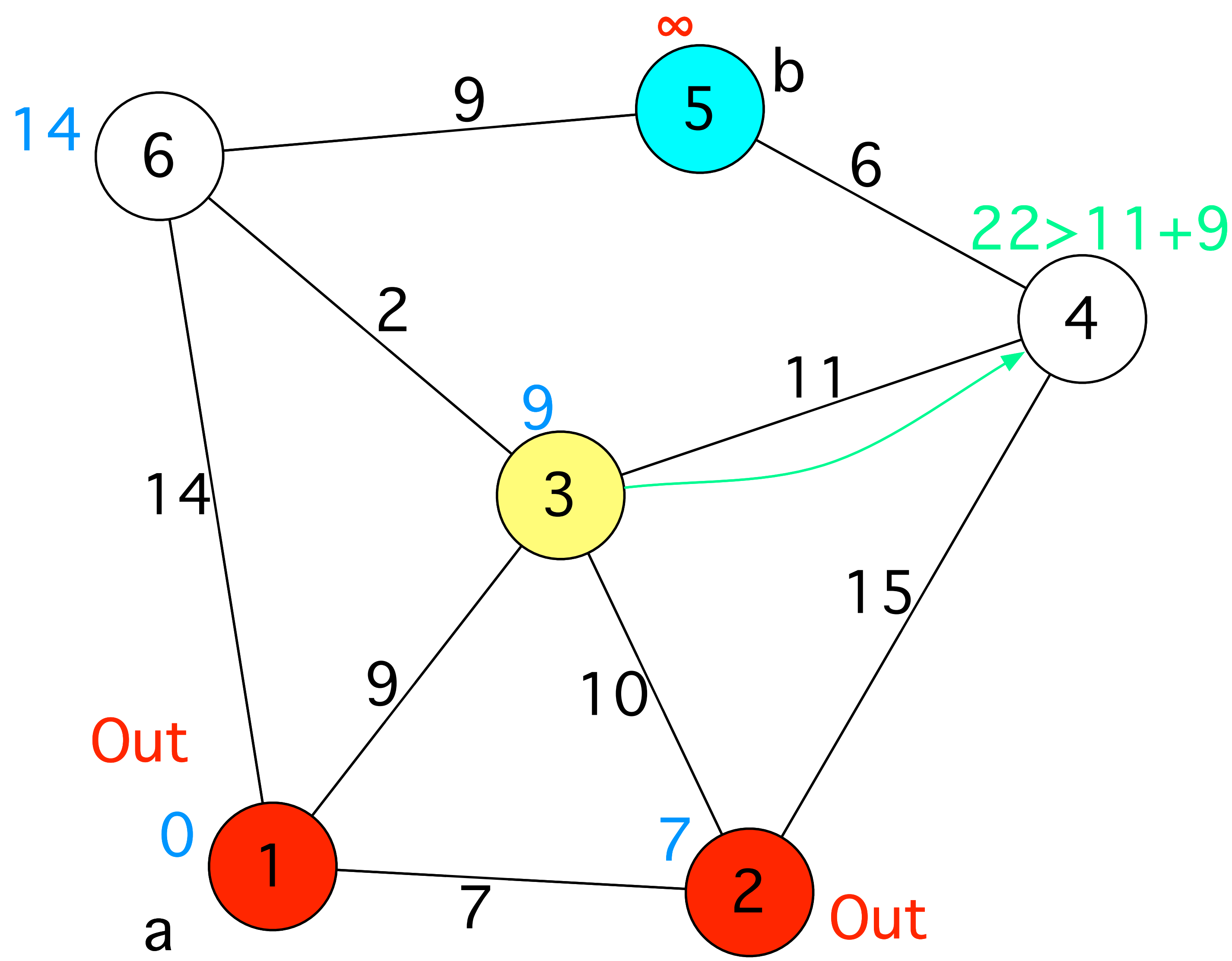
Graph Methods

Dijkstra's algorithm



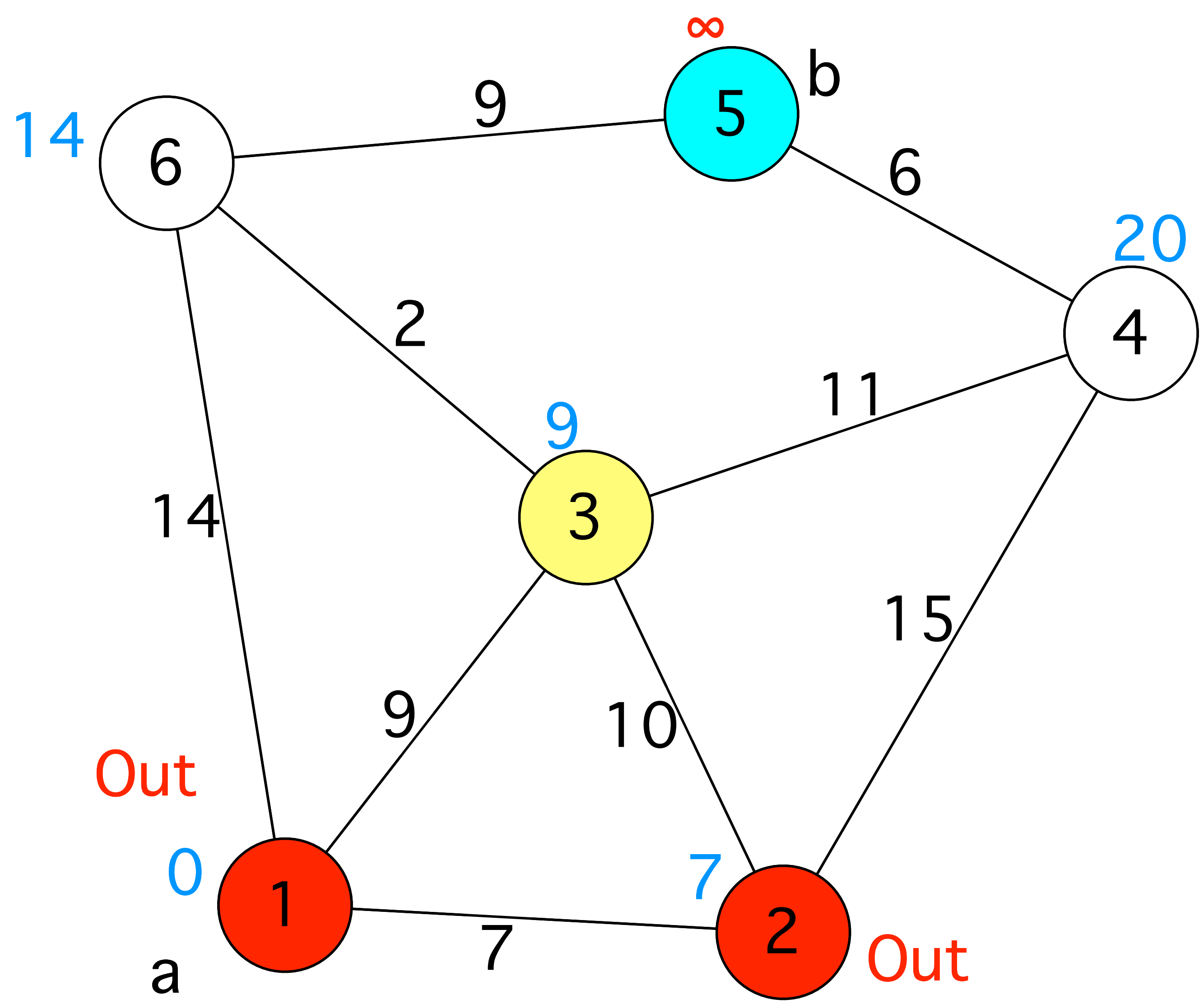
Graph Methods

Dijkstra's algorithm

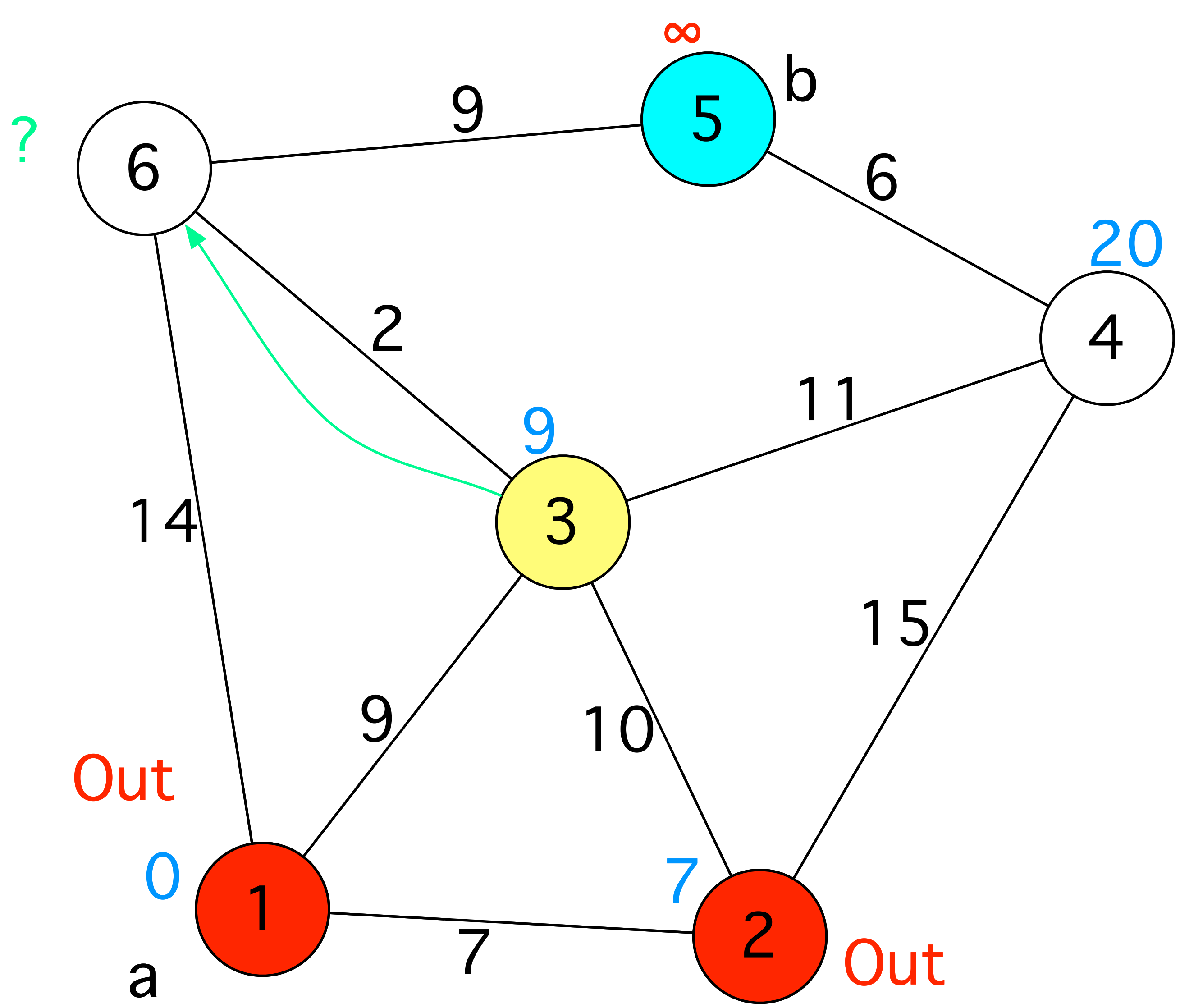


Graph Methods

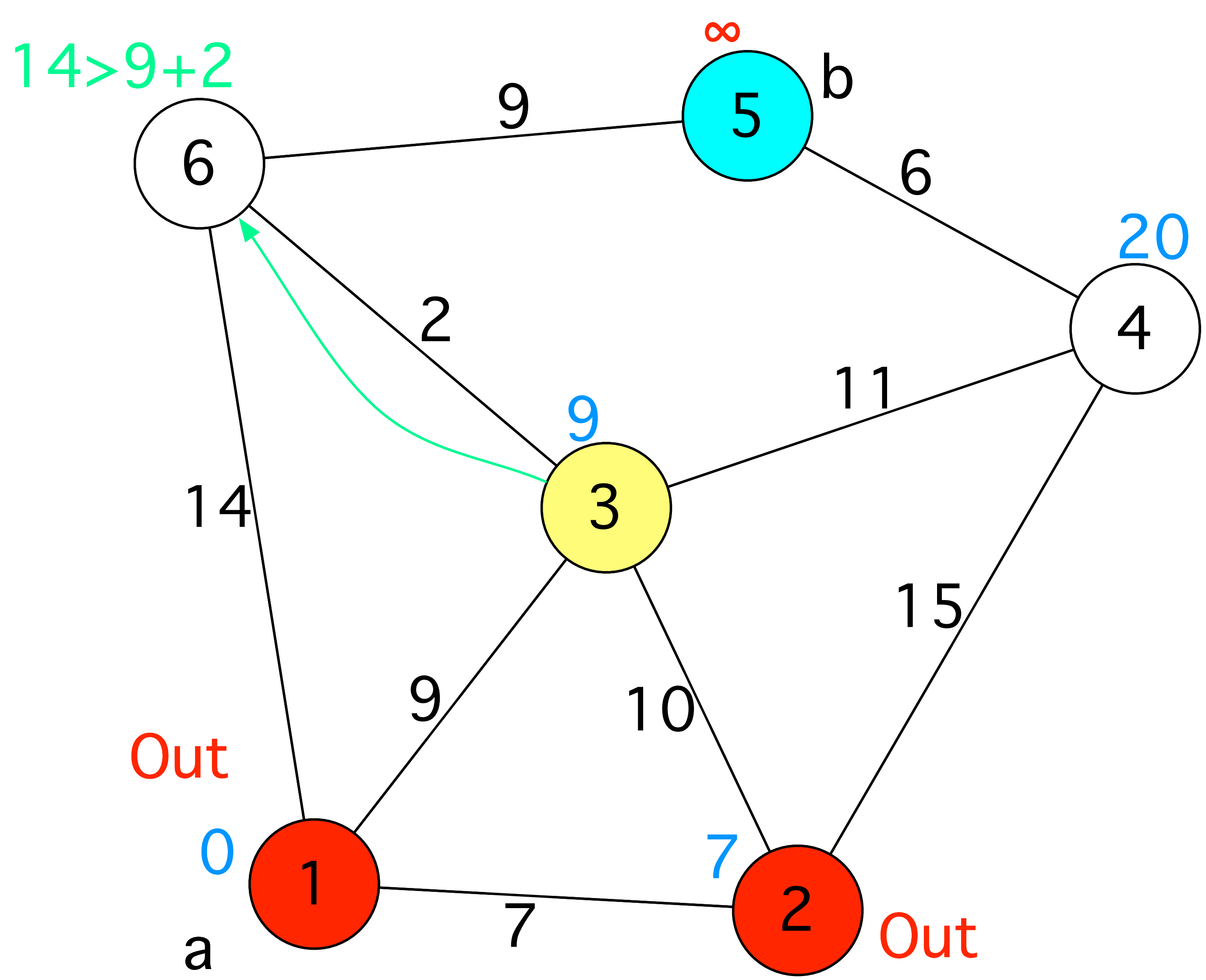
Dijkstra's algorithm



Graph Methods
Dijkstra's algorithm

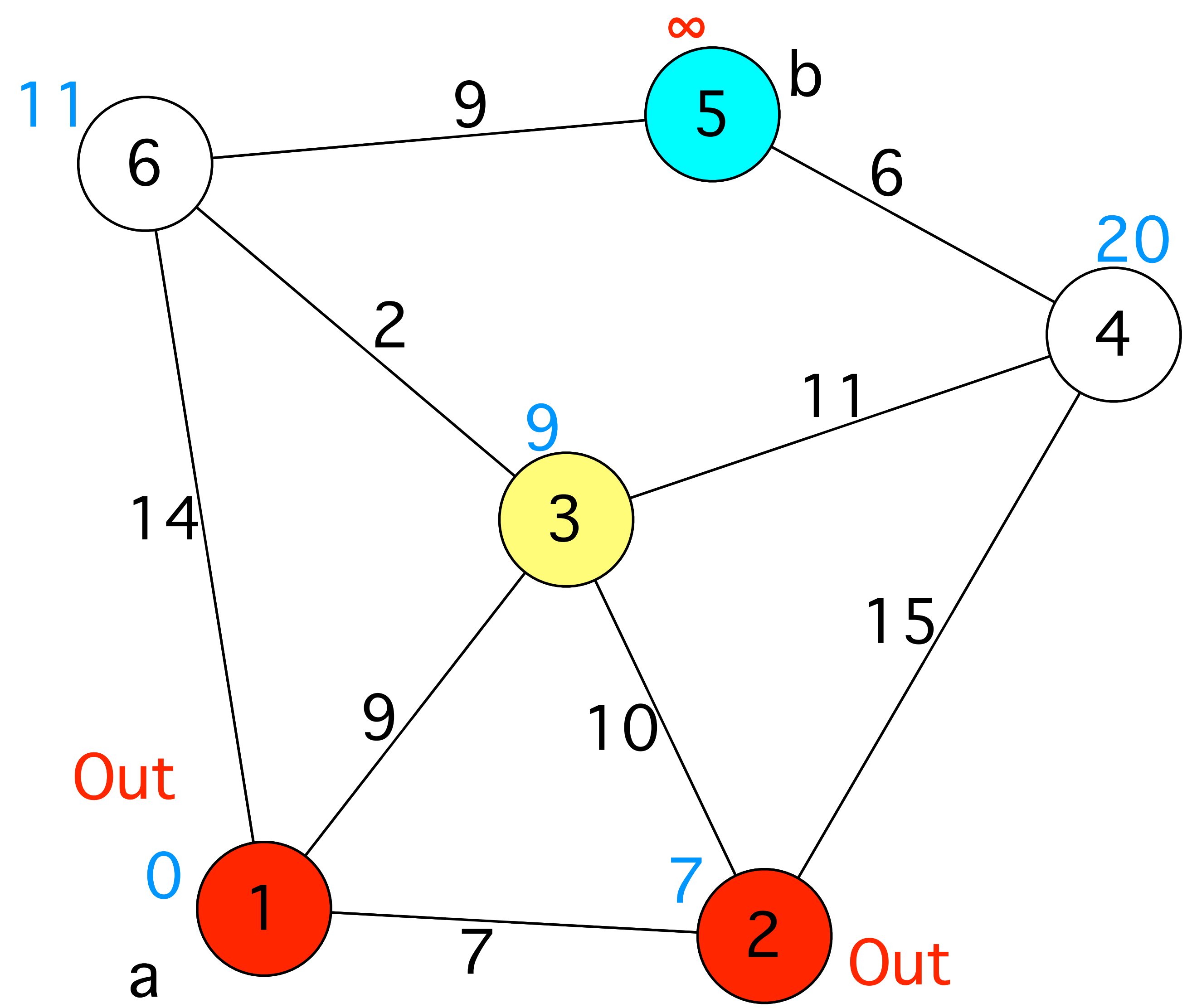


Graph Methods
Dijkstra's algorithm



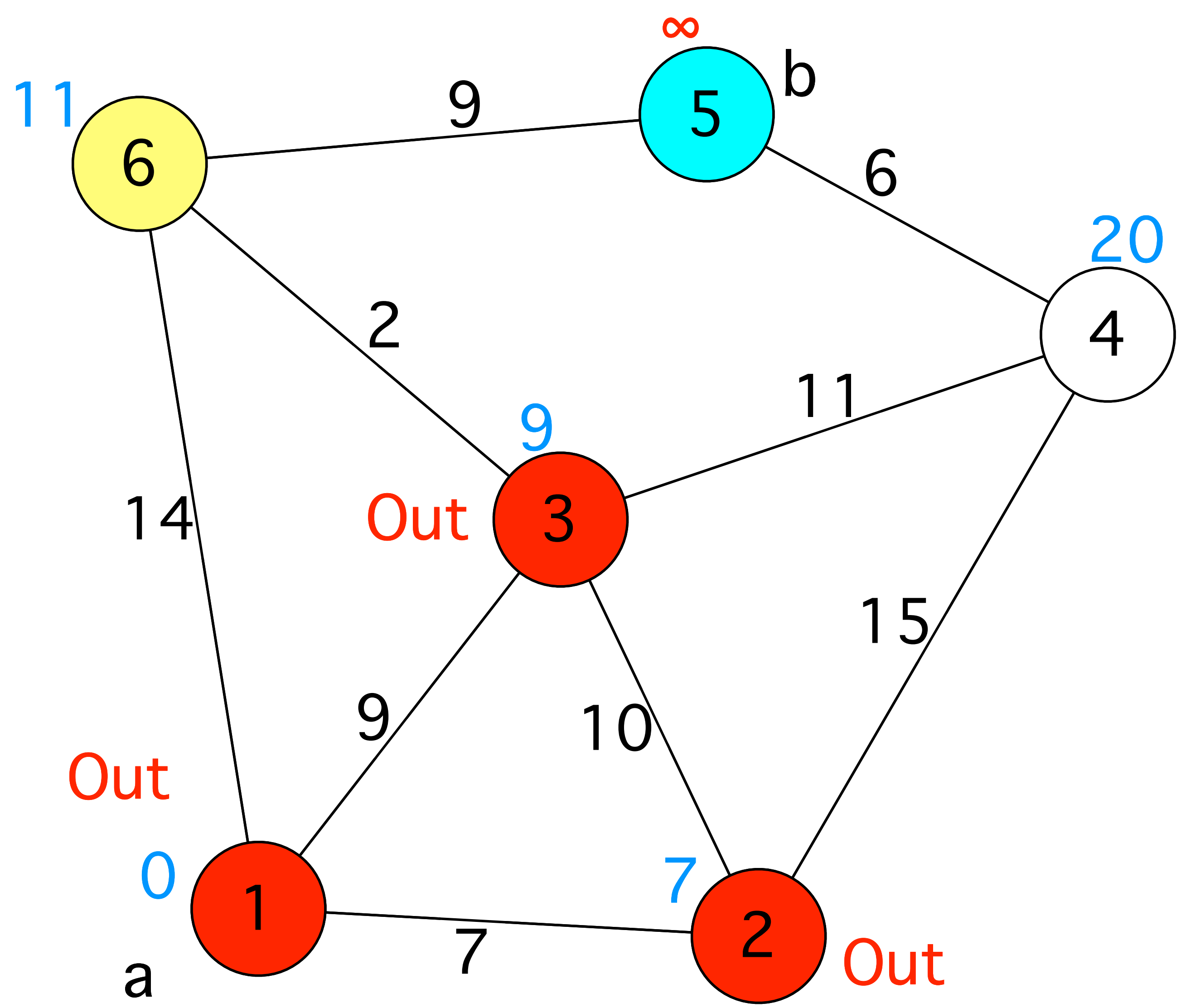
Graph Methods

Dijkstra's algorithm



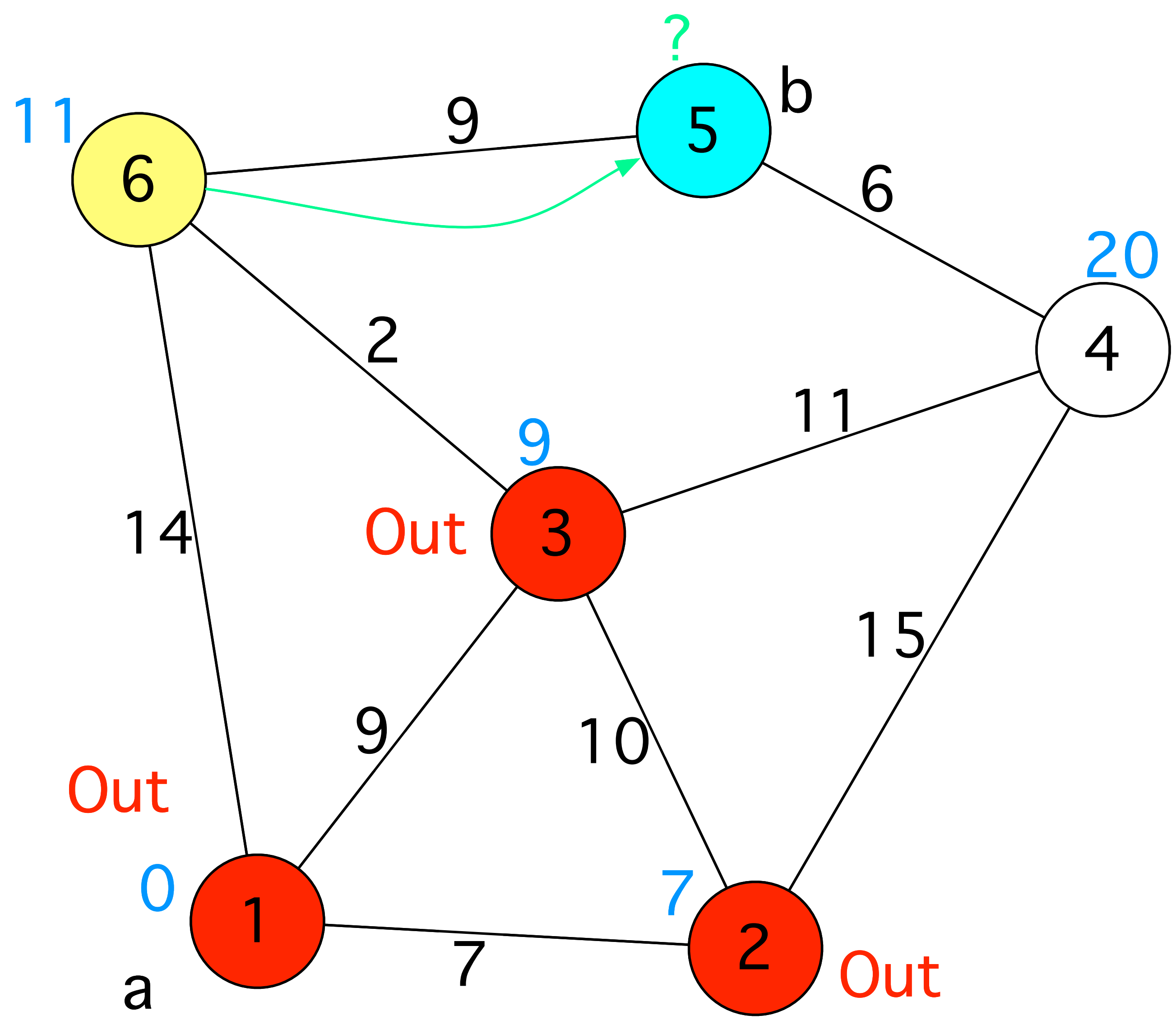
Graph Methods

Dijkstra's algorithm



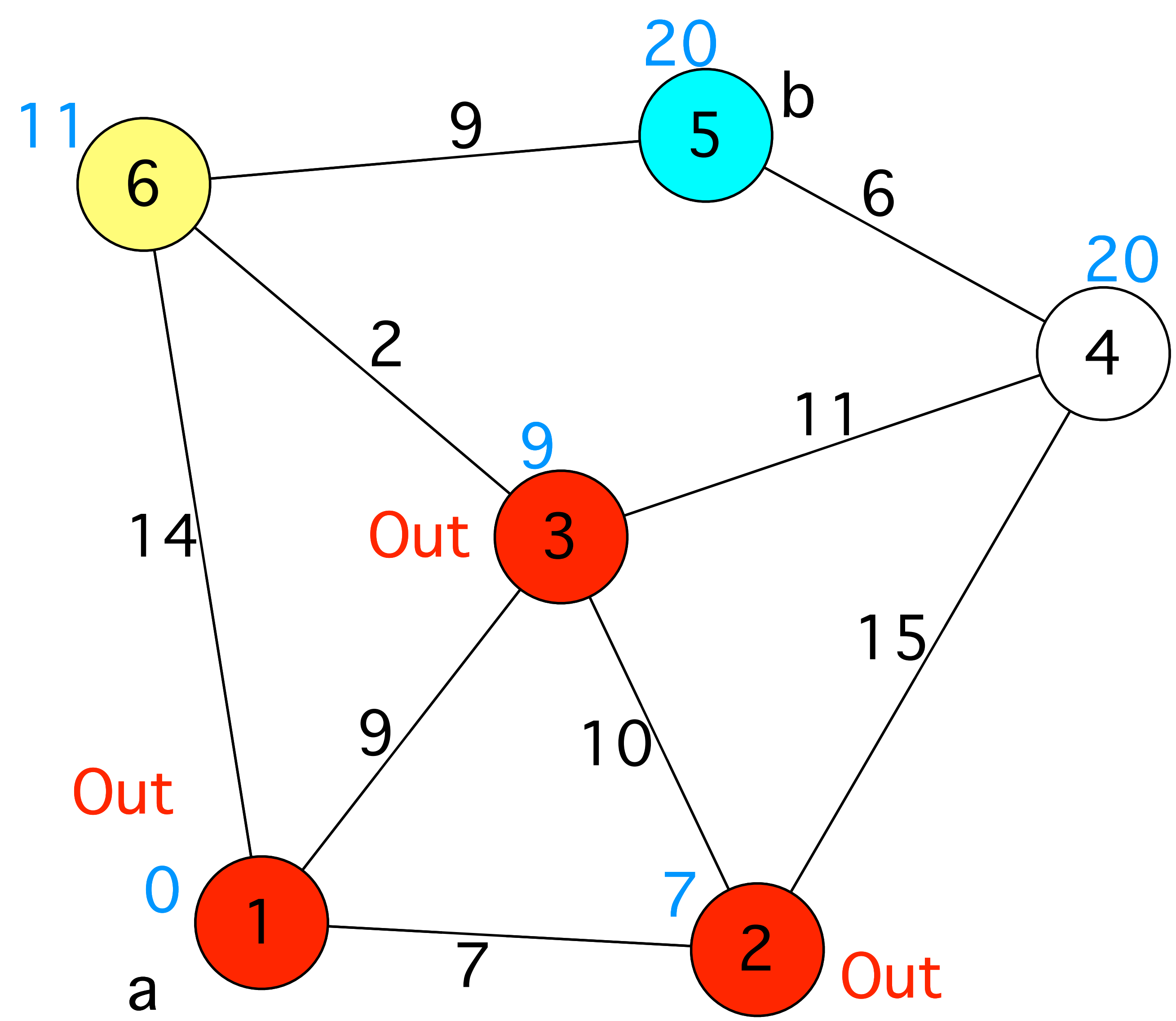
Graph Methods

Dijkstra's algorithm



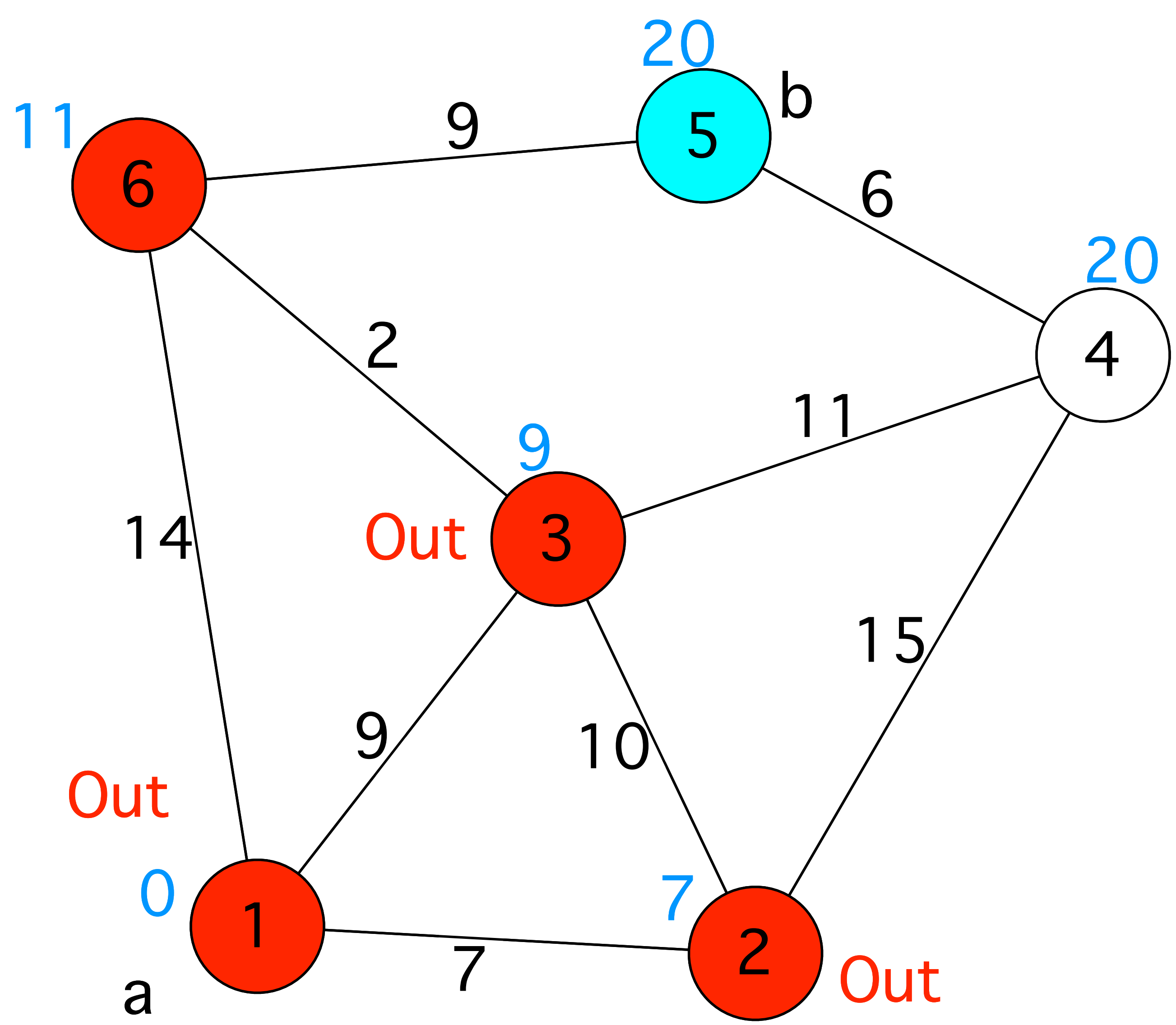
Graph Methods

Dijkstra's algorithm

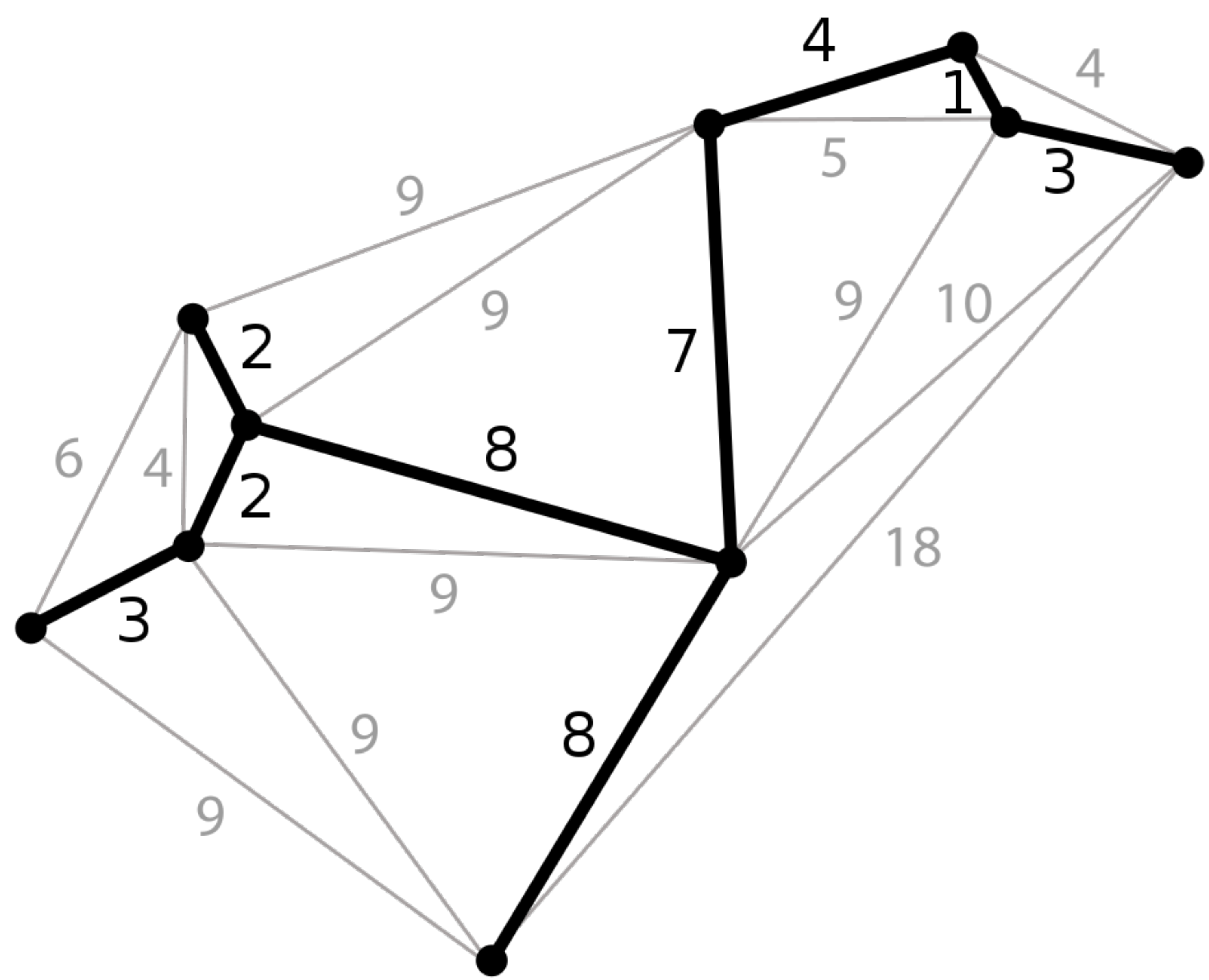


Graph Methods

Dijkstra's algorithm

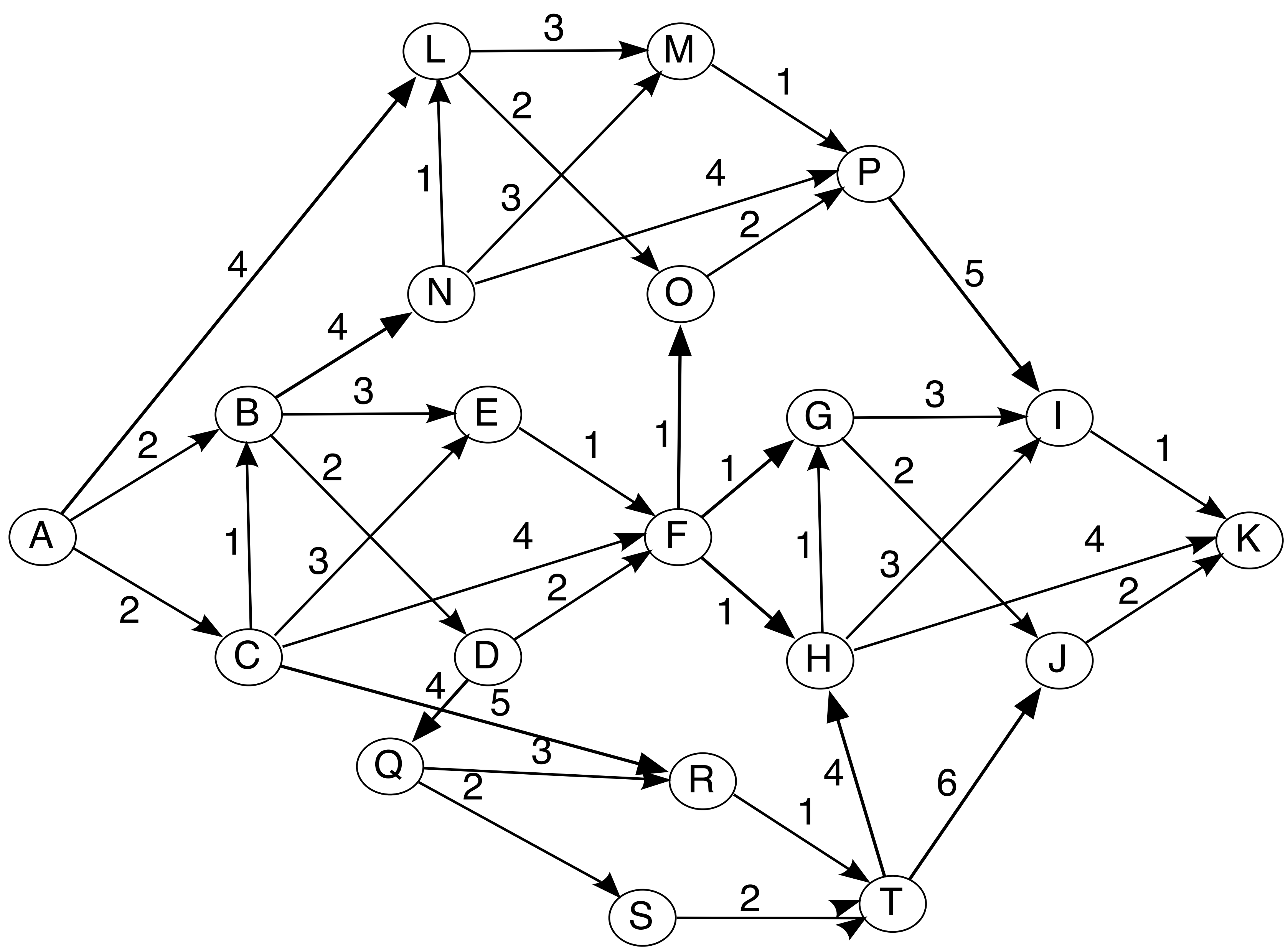


Graph Methods
minimum spanning tree



Graph Methods

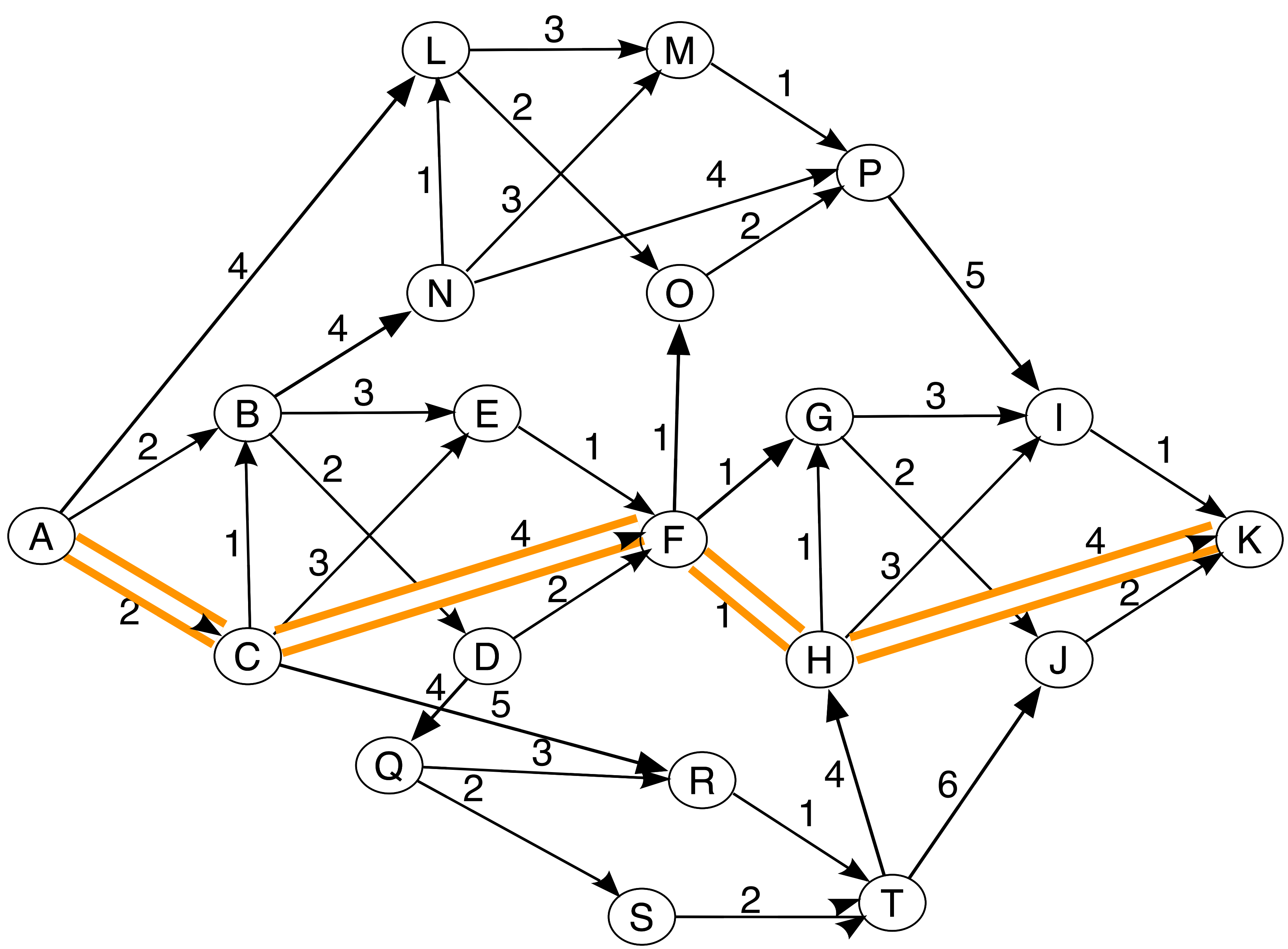
min cut/max flow



Social Playlists and Bottleneck Measurements:
Exploiting Musician Social Graphs Using Content-Based Dissimilarity and Pairwise Maximum Flow Values
Fields, Ben and Jacobson, Kurt and Rhodes, Christophe and Casey, Michael

Graph Methods

min cut/max flow



Social Playlists and Bottleneck Measurements:
Exploiting Musician Social Graphs Using Content-Based Dissimilarity and Pairwise Maximum Flow Values
Fields, Ben and Jacobson, Kurt and Rhodes, Christophe and Casey, Michael

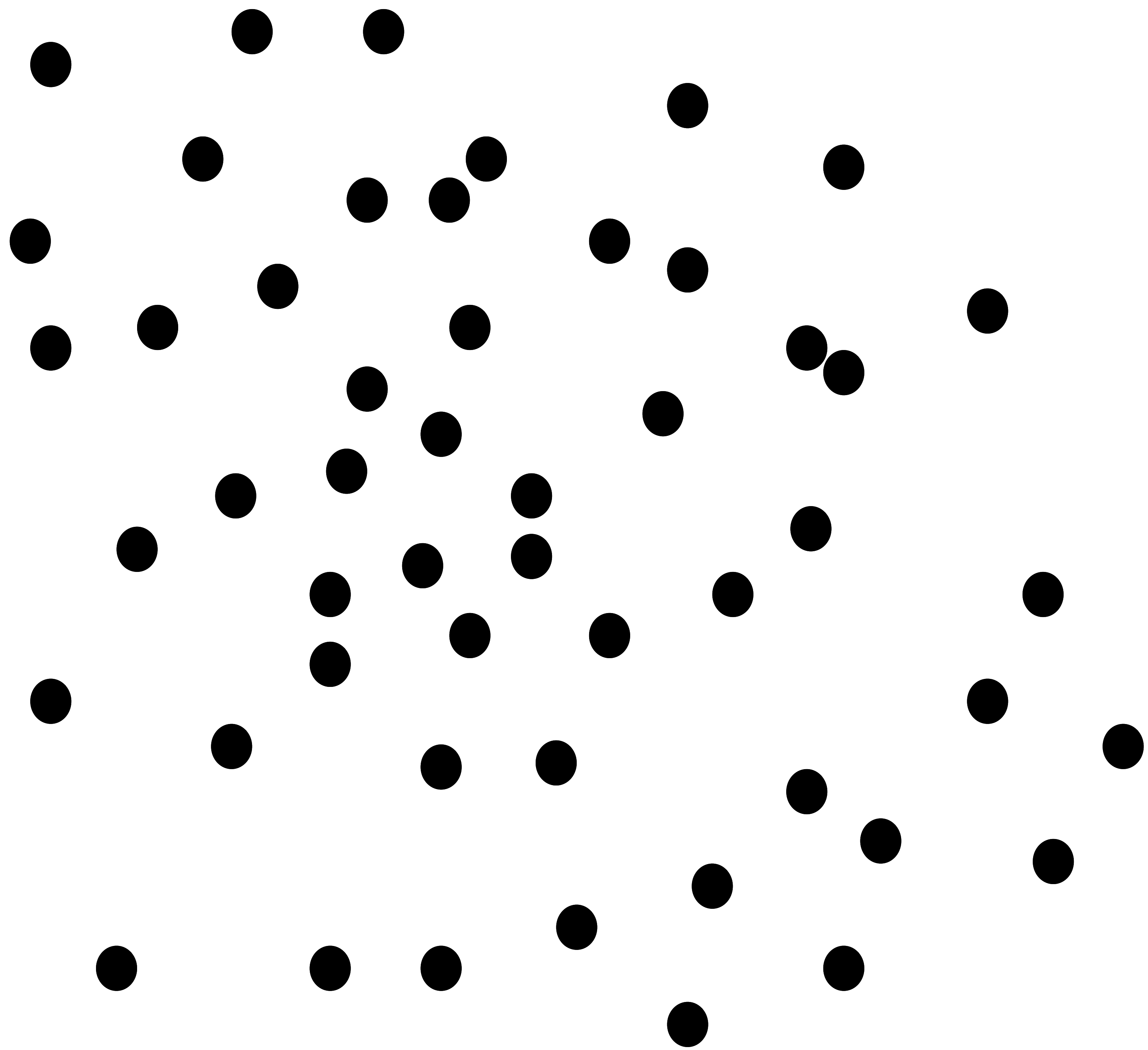
Graph-Based Path Finding

- A **directed graph** is created based on the **friend** connections amongst artists found on **myspace**
- The edges of this graph are weighted using content-based similarity
- Playlists are constructed through the use of the **max flow/min cut** from a starting to ending artist

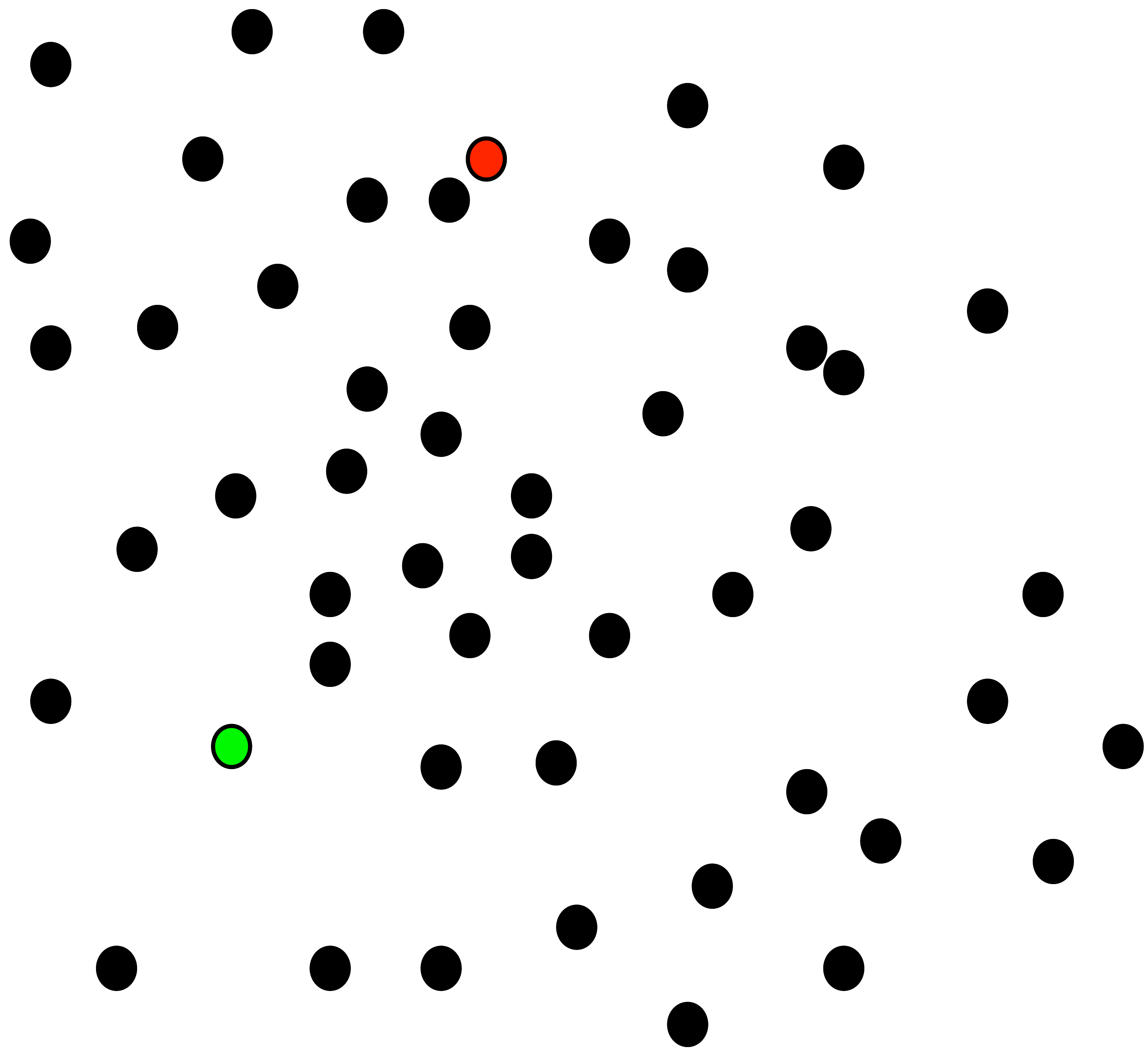
Break

Part II

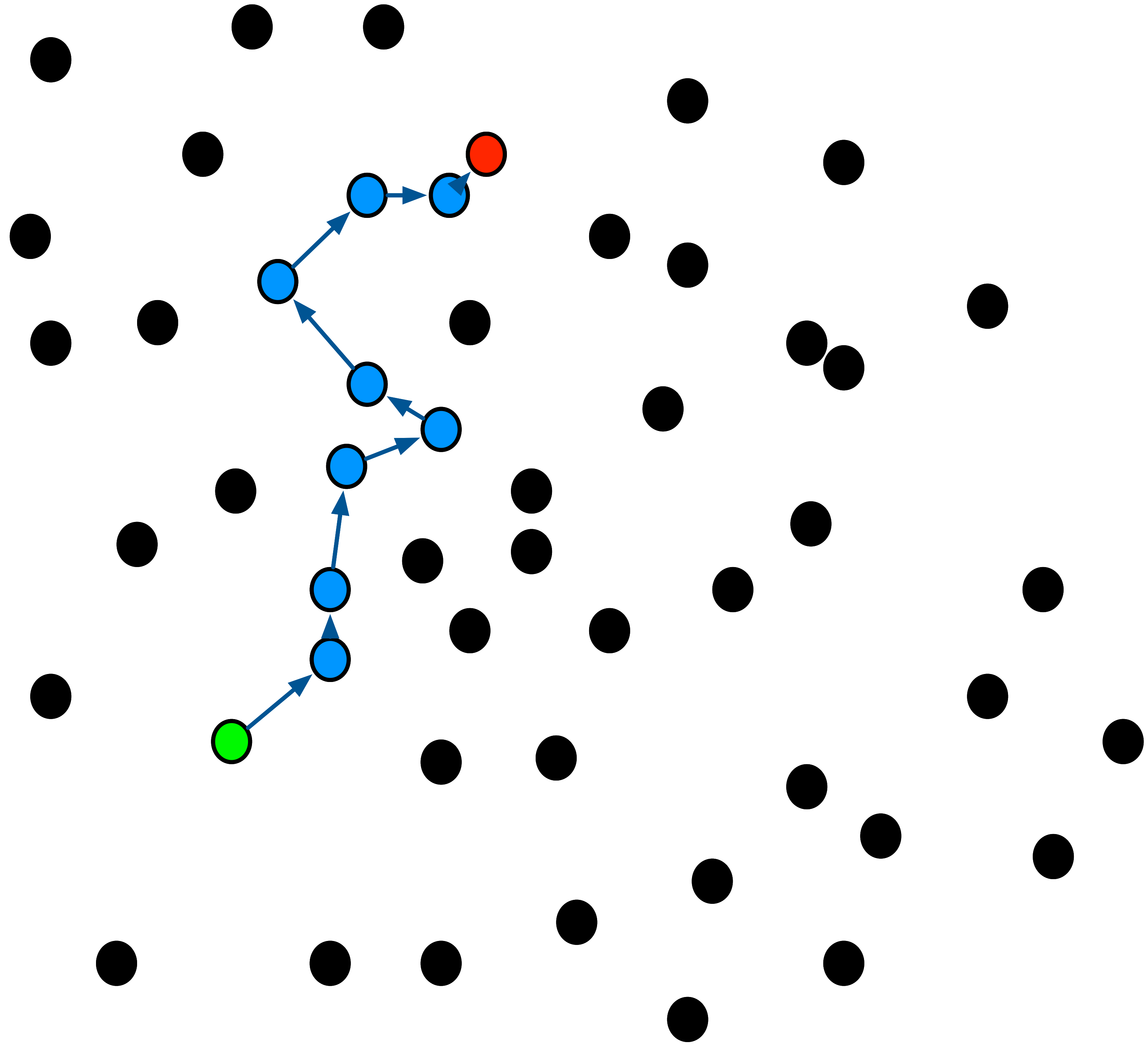
Points-In-Space



Points-In-Space



Points-In-Space



Start-End Timbrel Paths

1. For every song, calculate divergence from select start ($D_{KL}(i,s)$) and end ($D_{KL}(i,e)$) songs
2. Find *d*% songs with highest divergence from start song; repeat against end song. Remove songs that appear in both sets.
3. Compute divergent ratio for remaining songs: $$R(i) = \frac{D_{KL}(i,s)}{D_{KL}(i,e)}$$

Start-End Timbrel Paths

4. Compute ideal step width:

$$step = \frac{R(s) - R(e)}{p + 1}$$

5. Generate ideal positions for each song:

$$\hat{R}(j) = R(s) + j * step$$

6. Select ideal songs that best match the ideal:

$$S_j = \arg \min_{i=1,\ldots,m} |\hat{R}(j) - R(i)|$$

Evaluating S-E Paths
objective analysis

- The playlist should contain mostly songs from genres *A* and *B*
- At the beginning of the playlist, most songs should be from genre *A*, at the end from genre *B* and from both genres in the middle

Evaluating S-E Paths

objective analysis

	HiHo	Regg	Funk	Elec	Pop	**Rock**
Sec1	33	5	2	15	8	38
Sec2	5	1	2	7	4	81
Sec3	2	0	3	4	2	88

	HiHo	**Regg**	Funk	Elec	Pop	**Rock**
Sec1	26	7	2	20	7	38
Sec2	6	1	2	7	4	80
Sec3	3	0	2	4	2	88

	HiHo	Regg	Funk	**Elec**	Pop	Rock
Sec1	30	5	2	35	8	19
Sec2	6	2	3	66	5	18
Sec3	2	2	3	70	4	18

	HiHo	Regg	**Funk**	Elec	**Pop**	Rock
Sec1	19	3	8	28	13	29
Sec2	17	4	4	20	19	36
Sec3	12	3	4	22	16	42

Evaluating S-E Paths

objective analysis

	HiHo	Regg	Funk	Elec	Pop	**Rock**
Sec1	33	5	2	15	8	38
Sec2	5	1	2	7	4	81
Sec3	2	0	3	4	2	88

	HiHo	**Regg**	Funk	Elec	Pop	**Rock**
Sec1	26	7	2	20	7	38
Sec2	6	1	2	7	4	80
Sec3	3	0	2	4	2	88

	HiHo	Regg	Funk	**Elec**	Pop	Rock
Sec1	30	5	2	35	8	19
Sec2	6	2	3	66	5	18
Sec3	2	2	3	70	4	18

	HiHo	Regg	**Funk**	Elec	**Pop**	Rock
Sec1	19	3	8	28	13	29
Sec2	17	4	4	20	19	36
Sec3	12	3	4	22	16	42

Playlist Generation Using Start and End Songs
Arthur Flexer, Dominik Schnitzer, Martin Gasser and Gerhard Widmer

Evaluating S-E Paths

objective analysis

	HiHo	Regg	Funk	Elec	Pop	**Rock**
Sec1	33	5	2	15	8	38
Sec2	5	1	2	7	4	81
Sec3	2	0	3	4	2	88

	HiHo	**Regg**	Funk	Elec	Pop	**Rock**
Sec1	26	7	2	20	7	38
Sec2	6	1	2	7	4	80
Sec3	3	0	2	4	2	88

	HiHo	Regg	Funk	**Elec**	Pop	Rock
Sec1	30	5	2	35	8	19
Sec2	6	2	3	66	5	18
Sec3	2	2	3	70	4	18

	HiHo	Regg	**Funk**	Elec	**Pop**	Rock
Sec1	19	3	8	28	13	29
Sec2	17	4	4	20	19	36
Sec3	12	3	4	22	16	42

Playlist Generation Using Start and End Songs
Arthur Flexer, Dominik Schnitzer, Martin Gasser and Gerhard Widmer

Evaluating S-E Paths

objective analysis

	HiHo	Regg	Funk	Elec	Pop	**Rock**
Sec1	33	5	2	15	8	38
Sec2	5	1	2	7	4	81
Sec3	2	0	3	4	2	88

	HiHo	**Regg**	Funk	Elec	Pop	**Rock**
Sec1	26	7	2	20	7	38
Sec2	6	1	2	7	4	80
Sec3	3	0	2	4	2	88

	HiHo	Regg	Funk	**Elec**	Pop	Rock
Sec1	30	5	2	35	8	19
Sec2	6	2	3	66	5	18
Sec3	2	2	3	70	4	18

	HiHo	Regg	**Funk**	Elec	**Pop**	Rock
Sec1	19	3	8	28	13	29
Sec2	17	4	4	20	19	36
Sec3	12	3	4	22	16	42

Evaluating S-E Paths

objective analysis

	HiHo	Regg	Funk	Elec	Pop	**Rock**
Sec1	33	5	2	15	8	38
Sec2	5	1	2	7	4	81
Sec3	2	0	3	4	2	88

	HiHo	**Regg**	Funk	Elec	Pop	**Rock**
Sec1	26	7	2	20	7	38
Sec2	6	1	2	7	4	80
Sec3	3	0	2	4	2	88

	HiHo	Regg	Funk	**Elec**	Pop	Rock
Sec1	30	5	2	35	8	19
Sec2	6	2	3	66	5	18
Sec3	2	2	3	70	4	18

	HiHo	Regg	**Funk**	Elec	**Pop**	Rock
Sec1	19	3	8	28	13	29
Sec2	17	4	4	20	19	36
Sec3	12	3	4	22	16	42

Evaluating S-E Paths
subjective analysis

- How many outliers are in the playlist which do not fit the overall flavor of the playlist?
- Is the order of songs in the playlist from the start to the end song apparent?

Evaluating S-E Paths
subjective analysis

Genres from	Genres to	# of outliers	order apparent yes	order apparent somewhat	order apparent no
HiHo	Regg	4.7		x	xx
HiHo	Funk	1.7	xx	x	
HiHo	Elec	1.3	xxx		
HiHo	Pop	2.7		xx	x
HiHo	Rock	0	xxx		
Regg	Funk	0.7	xx	x	
Regg	Elec	1.3	xxx		
Regg	Pop	1.3	xxx		
Regg	Rock	0.3	xx		x
Funk	Elec	1.0	xx	x	
Funk	Pop	1.7	xx		x
Funk	Rock	0	xx	x	
Elec	Pop	0	xxx		
Elec	Rock	0	xx	x	
Pop	Rock	0	xxx		
average		1.1	71.1%	17.8%	11.1%

Evaluating S-E Paths

subjective analysis

Genres from	Genres to	# of outliers	order apparent yes	order apparent somewhat	order apparent no
HiHo	Regg	4.7		x	xx
HiHo	Funk	1.7	xx	x	
HiHo	Elec	1.3	xxx		
HiHo	Pop	2.7		xx	x
HiHo	Rock	0	xxx		
Regg	Funk	0.7	xx	x	
Regg	Elec	1.3	xxx		
Regg	Pop	1.3	xxx		
Regg	Rock	0.3	xx		x
Funk	Elec	1.0	xx	x	
Funk	Pop	1.7	xx		x
Funk	Rock	0	xx	x	
Elec	Pop	0	xxx		
Elec	Rock	0	xx	x	
Pop	Rock	0	xxx		
	average	1.1	71.1%	17.8%	11.1%

Evaluating S-E Paths

subjective analysis

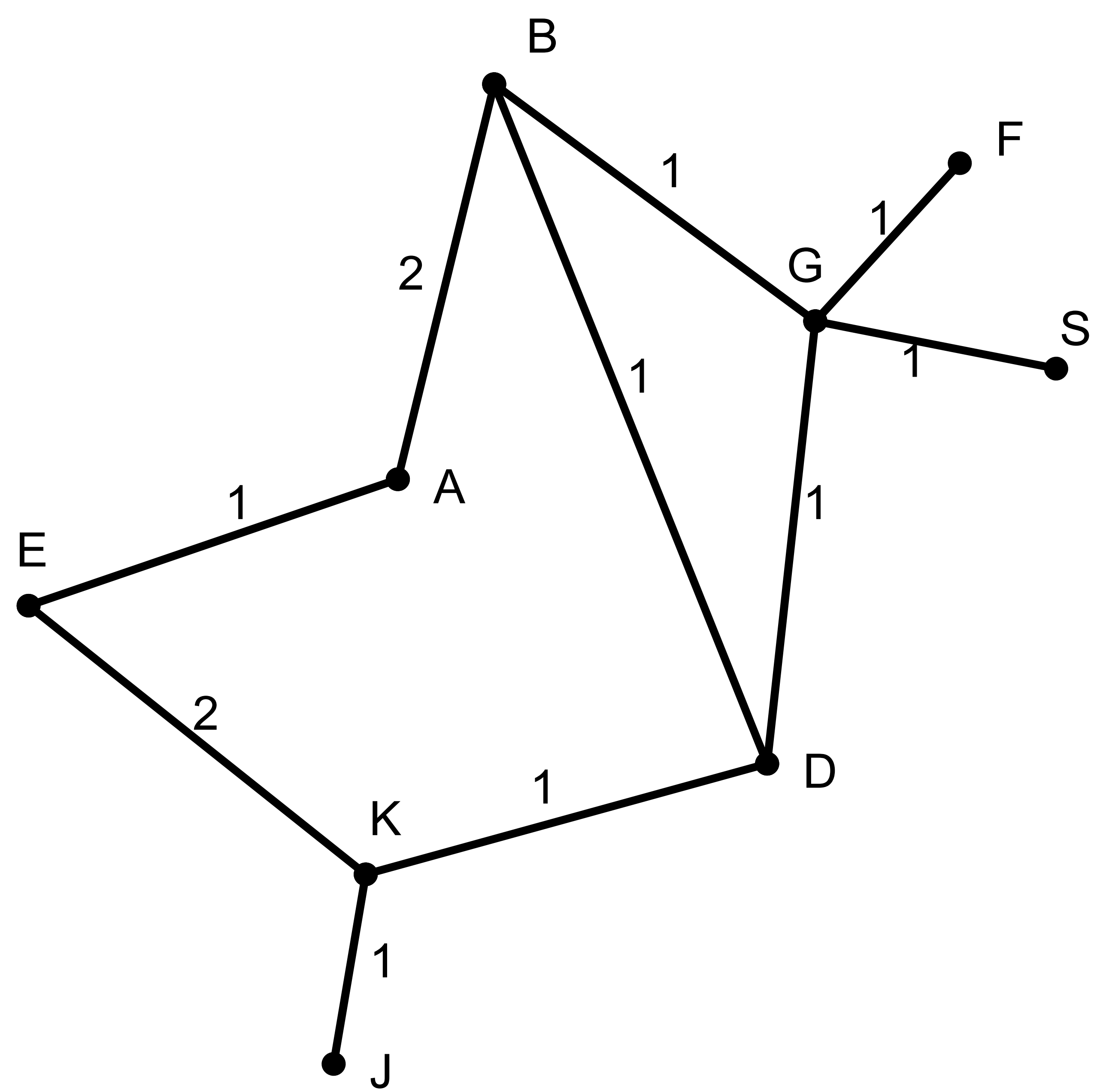
Genres		# of outliers	order apparent		
from	to		yes	somewhat	no
HiHo	Regg	4.7		x	xx
HiHo	Funk	1.7	xx	x	
HiHo	Elec	1.3	xxx		
HiHo	Pop	2.7		xx	x
HiHo	Rock	0	xxx		
Regg	Funk	0.7	xx	x	
Regg	Elec	1.3	xxx		
Regg	Pop	1.3	xxx		
Regg	Rock	0.3	xx		x
Funk	Elec	1.0	xx	x	
Funk	Pop	1.7	xx		x
Funk	Rock	0	xx	x	
Elec	Pop	0	xxx		
Elec	Rock	0	xx	x	
Pop	Rock	0	xxx		
average		1.1	71.1%	17.8%	11.1%

Playlist Similarity

- The co-occurrence of objects in an **authored stream** can be used as a proxy for object similarity
- This sort of similarity is especially effective for the generation of playlists
- Employs the use of an undirected graph, weighted by co-occurrence counts

Inferring similarity between music objects with application to playlist generation
R. Ragno and C.J.C. Burges and C. Herley

Playlist Similarity



Inferring similarity between music objects with application to playlist generation
R. Ragno and C.J.C. Burges and C. Herley

Playlist Similarity
example playlists

Paperback Writer [Beatles]	0.0
Breakfast In America [Supertramp]	8.607
We're An American Band [Grand Funk Rrd]	8.607
In The Dark [Billy Squier]	17.244
I Shot The Sheriff [Eric Clapton]	12 .192
Fat Bottomed Girls [Queen]	16.335
Jumpin' Jack Flash [Rolling Stones]	13.723
Working For The Weekend [Loverboy]	15.251
Dream Weaver [Gary Wright]	15.520
Smells Like Teen Spirit! [Nirvana]	15.735
Can't Stop [Red Hot Chili Peppers]	16.732
Still Waiting [Sum 41]	19.256
Grave Digger [Dave Matthews]	20. 665

Lithium [Nirvana] : 0.0

Fall To Pieces [Velvet Revolver]	7.668
Tonight, Tonight [Smashing Pumpkins]	12.712
Slow Hands [Interpol]	12.712
Renegades Of Funk [Rage Against...]	10.127
Before I Forget [Slipknot]	7.355
The Kids Aren't Alright [Offspring]	11.712
All These Things That I've Done [Killers]	9.542
Weapon [Matthew Good]	18.914
Kryptonite [3 Doors Down]	11.127
Home [Three Days Grace]	8.712
Whatever [Godsmack]	10.127
Colors [Crossfade]	7.097

Inferring similarity between music objects with application to playlist generation
R. Ragno and C.J.C. Burges and C. Herley

Playlist Similarity

example similarities

Hey Jude [Beatles]	0.000
Lady Madonna [Beatles]	7.515
Lucy In The Sky With Diamonds [Beatles]	7.515
Peace Of Mind [Boston]	7.737
(Just Like) Starting Over [John Lennon]	7.737
Saturday In The Park [Chicago]	8.000
Shine It All Around [Robert Plant]	8.000
Holiday [Green Day]	8.000
Rock And Roll Heaven [Righteous Brothers]	8.000

Highway To Hell [AC/DC]	0.000
Best Of You [Foo Fighters]	6.252
Remedy [Seether]	6.362
Right Here [Staind]	6.362
Holiday [Green Day]	6.362
Be Yourself [Audioslave]	6.5 58
The Hand That Feeds [Nine Inch Nail s]	6.584
B.Y.O.B. [System Of A Down]	6.754
Happy? [Mudvayne]	6.847
Shine It All Around [Robert Plant]	6.982

Inferring similarity between music objects with application to playlist generation
R. Ragno and C.J.C. Burges and C. Herley

Playlist Steering

- Create a timbrel features
- Create the space using tuple and triple n-gram sequences from playlist logs
- Generate playlists via tag steering

Steerable Playlist Generation by Learning Song Similarity from Radio Station Playlists
Maillet, François and Eck, Douglas and Desjardins, Guillaume and Lamere, Paul

Playlist Steering

1. Select a seed track
2. Threshold transition matrix to generate set of possible next tracks
3. User creates a tag cloud, assigning weights to any of 360 tags
4. Autotagger creates tag cloud for all candidate tracks selected in (2). Cosine distance is taken between the user's tag cloud and each song's.
5. The track with the minimum cosine distance from seed is played

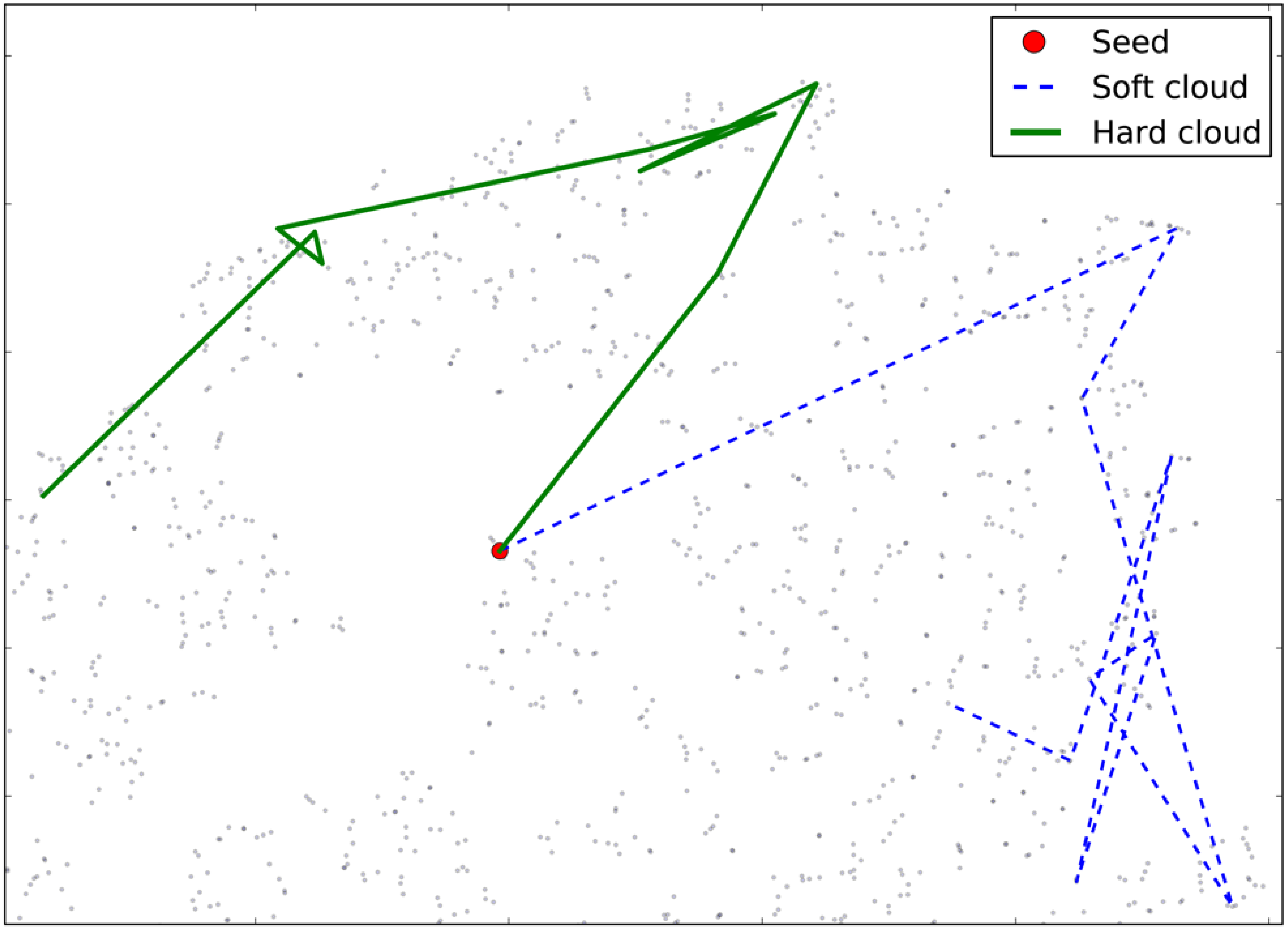
Steerable Playlist Generation by Learning Song Similarity from Radio Station Playlists
Maillet, François and Eck, Douglas and Desjardins, Guillaume and Lamere, Paul

Playlist Steering

Soft tag cloud
Viva la Vida by Coldplay
Wish You Were Here by Pink Floyd
Peaceful, Easy Feeling by Eagles
With or Without You by U2
One by U2
Fields Of Gold by Sting
Every Breath You Take by The Police
Gold Dust Woman by Fleetwood Mac
Enjoy The Silence by Depeche Mode

Hard tag cloud
All I Want by Staind
Re-Education (Through Labor) by Rise Against
Hammerhead by The Offspring
The Kill by 30 Seconds To Mars
When You Were Young by The Killers
Hypnotize by System of a Down
Breath by Breaking Benjamin
My Hero by Foo Fighters
Turn The Page by Metallica

Playlist Steering



Steerable Playlist Generation by Learning Song Similarity from Radio Station Playlists
Maillet, François and Eck, Douglas and Desjardins, Guillaume and Lamere, Paul

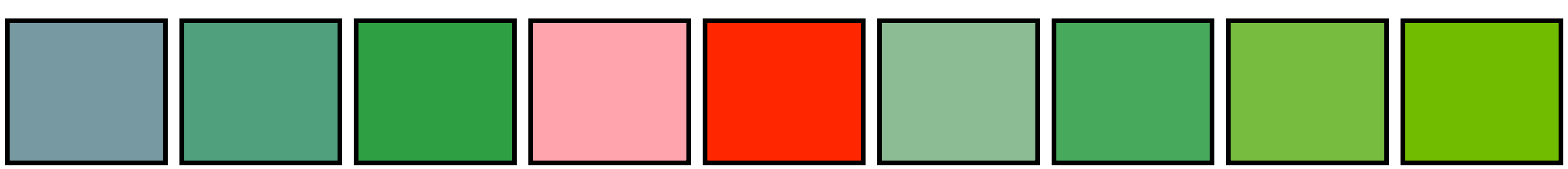
Scaling up playlisting

Scaling up playlist generation

- Building playlists involves satisfying constraints.
    - Global constraints: no duplicate songs, tempo between 120 and 130 BPM
    - Ordering constraints: no consecutive artists, DMCA rules
    - Sorting constraints: ordered by danceability and loudness
    - Playlist length: 15 songs, 32 minutes, < 20mb
- Finite constraint satisfaction problem. It's NP-HARD

General Approach

- Playlist is a sequence of songs: *`S1, S2 ... Sn`* drawn from a large pool of songs
- `Cost(Sn, C)` is how well song S at position N satisfies constraint C
- `Cost(Sn)` is total cost for song S at position N for all constraints
- `Cost(P)` is total cost of all songs in the Playlist
- Goal: **Find *S1*, ... *Sn* that minimizes Cost(P)**



Scaling up playlist generation

```
Generate random playlist

while Cost(P) > threshold:
    Calculate Cost(Sn) for each song
    find max( Cost(Sn) ) that is not tabu
    find best possible replacement
```

worst variables for which no value can be found to decrease the total cost are labelled as **tabu** for a given number of iterations.

Typical runtime: 1.4 seconds for 10 song playlist from a pool of 20,000 songs with 10 constraints

Fast Generation of Optimal Music Playlists using Local Search

- Simulated annealing
- Heuristic improvements
  - Song domain reduction
  - Two level search:
    1. Replace, Insert, Delete
    2. Swap
  - Partial constraint voting

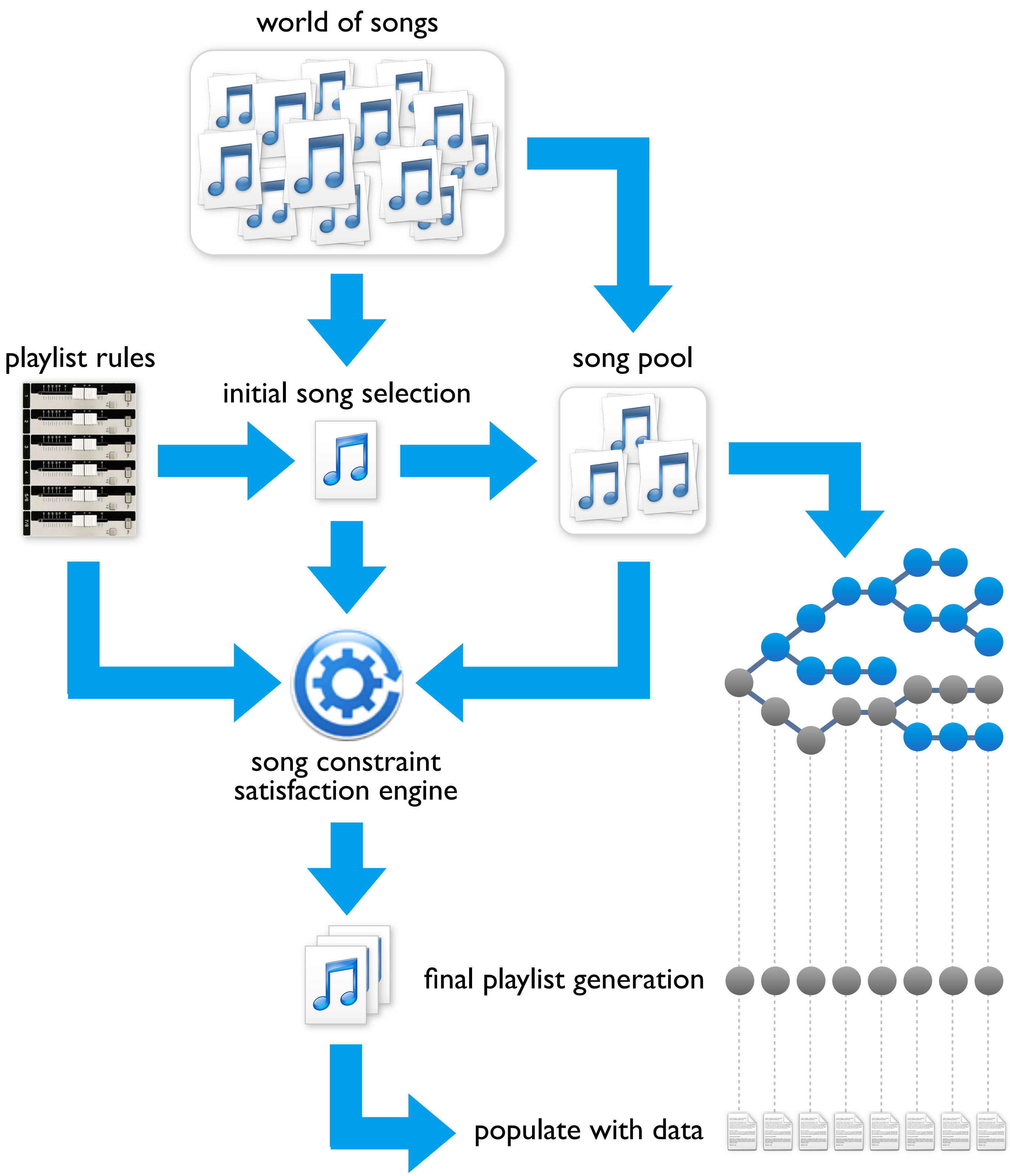
```
INITIALIZE p, t_0, L_0;
h := 0;
r := 0;
repeat
    for l := 1 to L_h do
    begin
        if r < β then
        begin
            if δ > random[0, 1) then
                GENERATE RANDOM p' ∈ N_reselect(p)
            else
                GENERATE p' ∈ N_reselect(p) BY VOTING;
            if f(p') ≤ f(p) or exp((f(p) − f(p'))/t) > random[0, 1)
            then p := p';
            r := r + 1
        end
        else begin
            p := NDR(p, γ);
            r := 0
        end
    end;
    h := h + 1;
    CALCULATE LENGTH L_h;
    CALCULATE CONTROL t_h
until STOP CRITERION
```

Fast Generation of Optimal Music Playlists using Local Search

- Start at high temperature
- Repeat until stop criterion:
  - Alternate - 100 at a time:
    - Level 1:
      - Select ***Replace***, ***Insert*** *or* ***Delete*** operation by **voting** or **random**
      - Create new playlist by applying the operation
      - Accept the change if:
        - It lowers the overall cost *or*
        - Randomly when at high temperatures
      - Lower the temperature
    - Level 2:
      - Non-deteriorating ***swap***

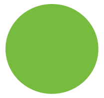
Typical runtime: 2 seconds for 14 song playlist with 15 constraints from a pool of 2,000 songs

The Echo Nest playlister

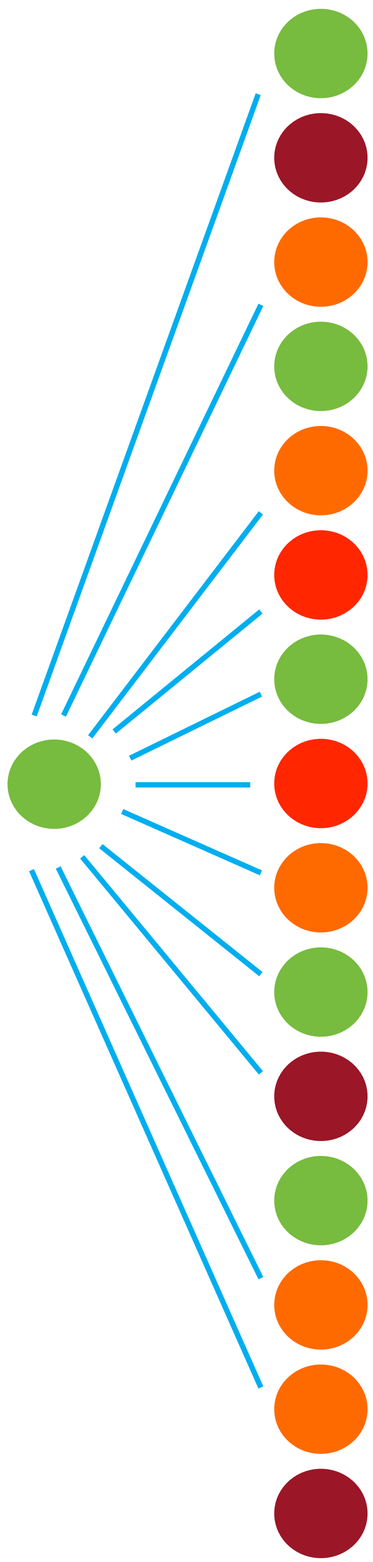


- Start with millions of songs
- Apply global constraints to create smaller song pool (1K to 10K songs)
- Use constraint engine to find best playlist:
  - Beam search
  - Adaptive search
- Populate with data

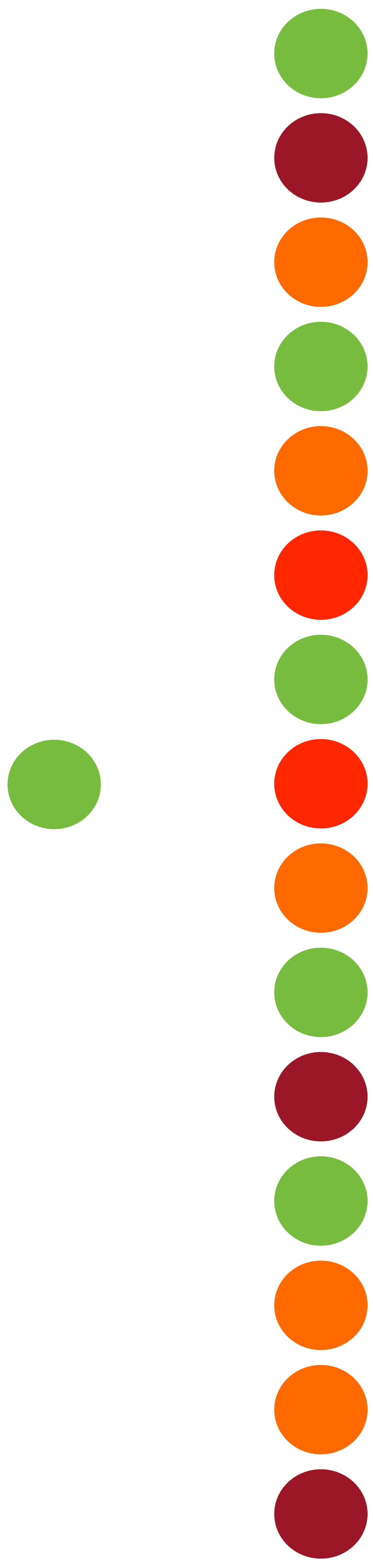
Beam Search



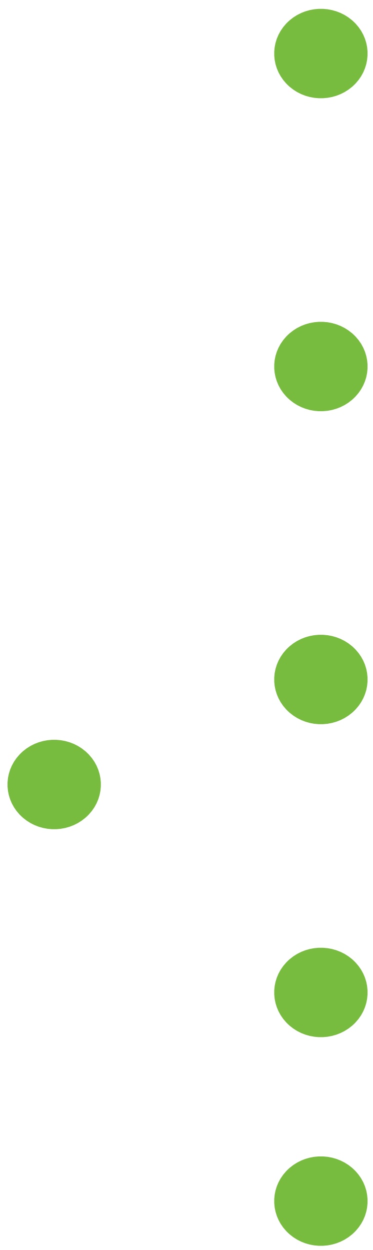
Beam Search



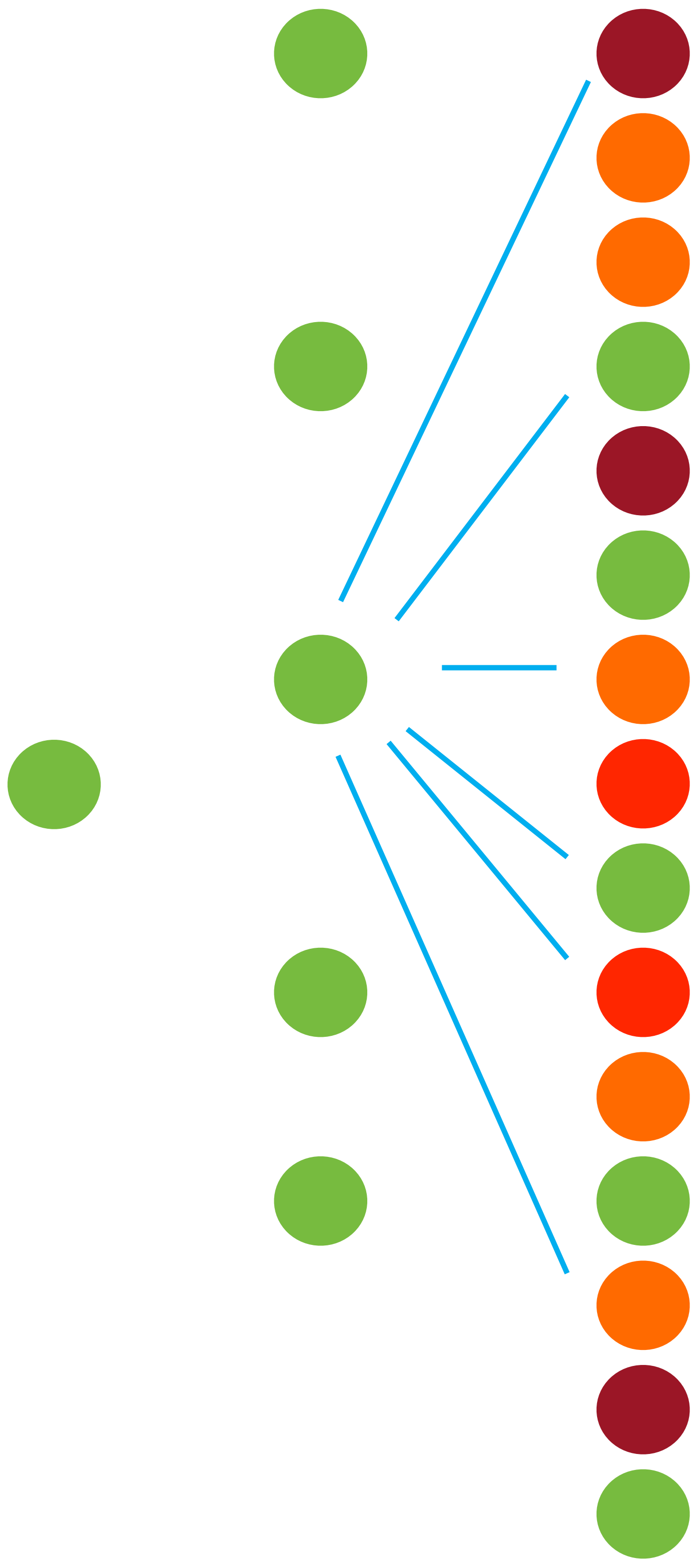
Beam Search



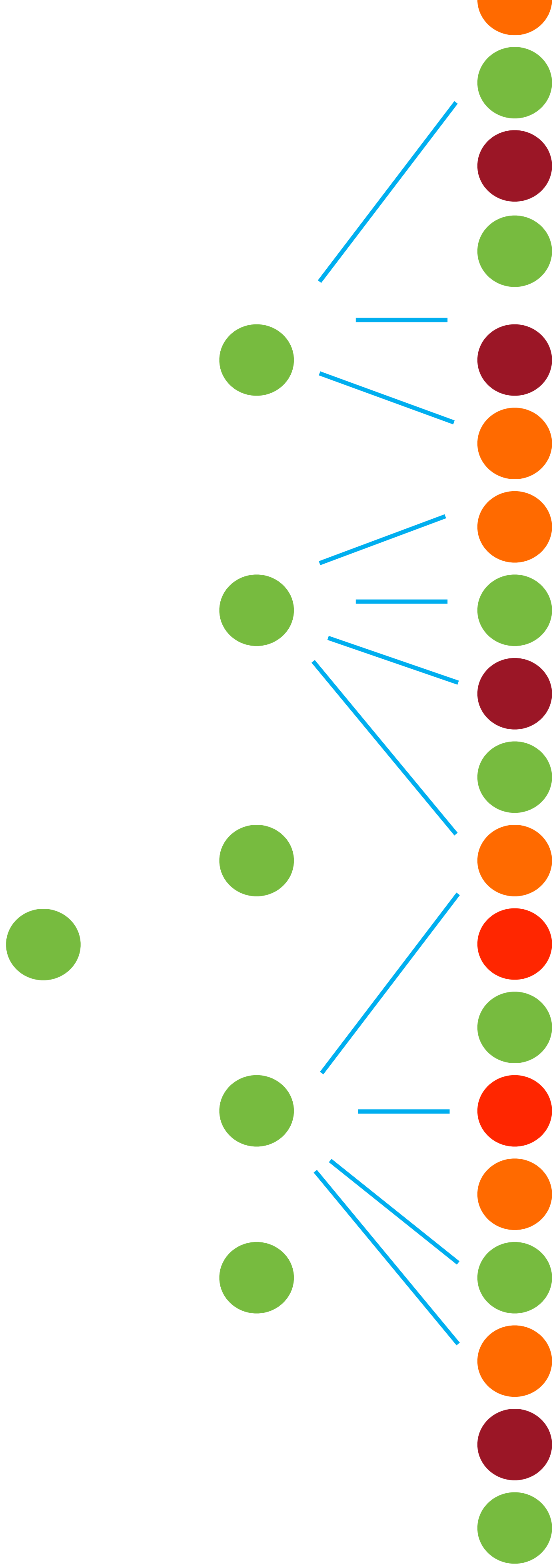
Beam Search



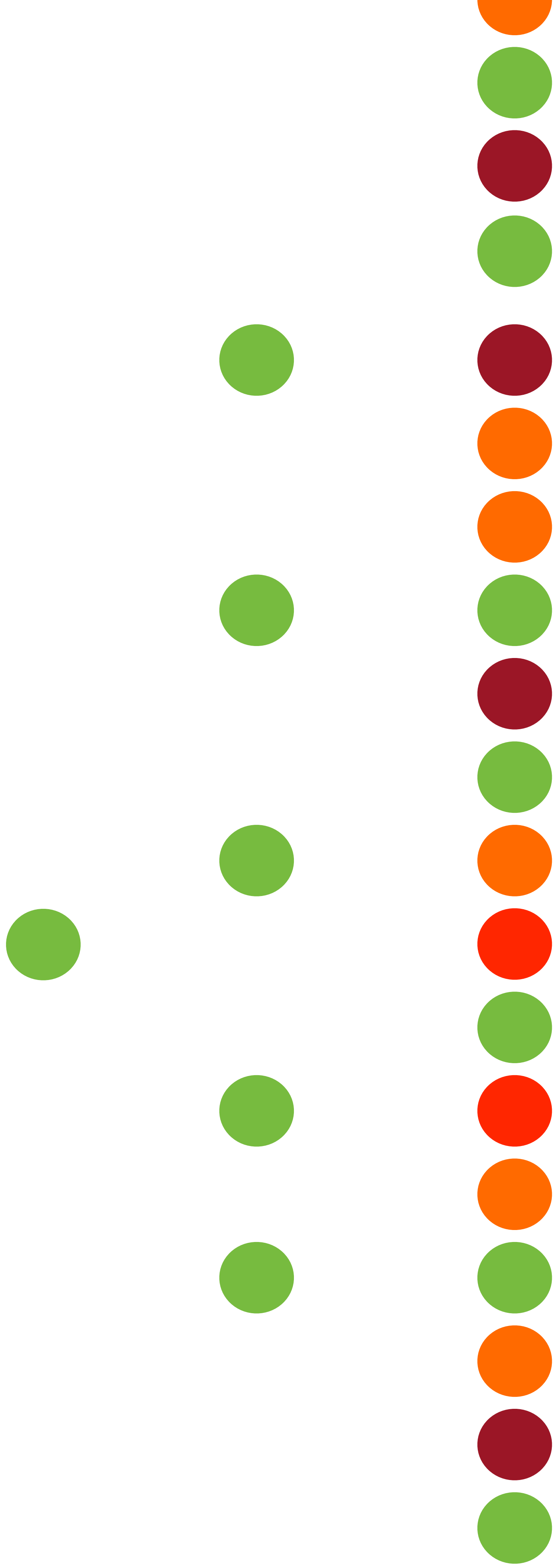
Beam Search



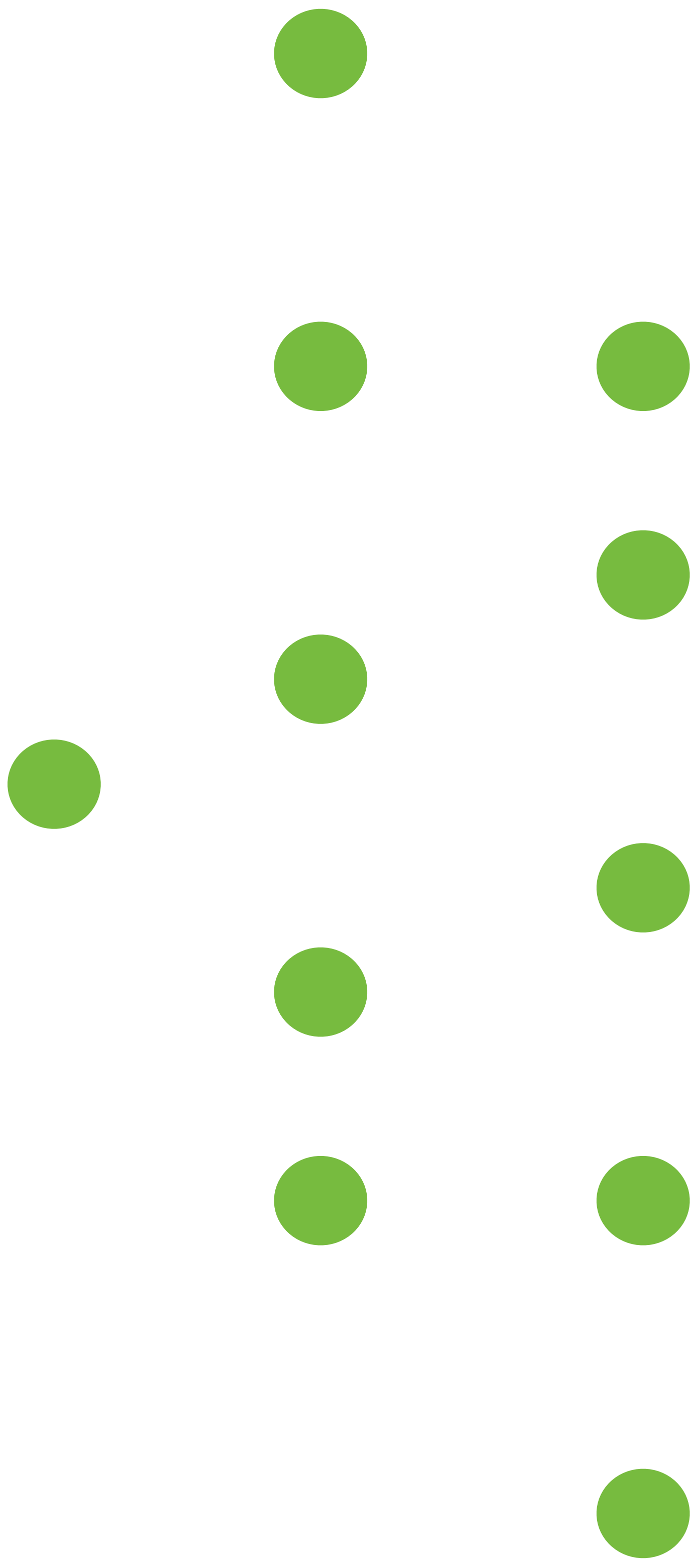
Beam Search



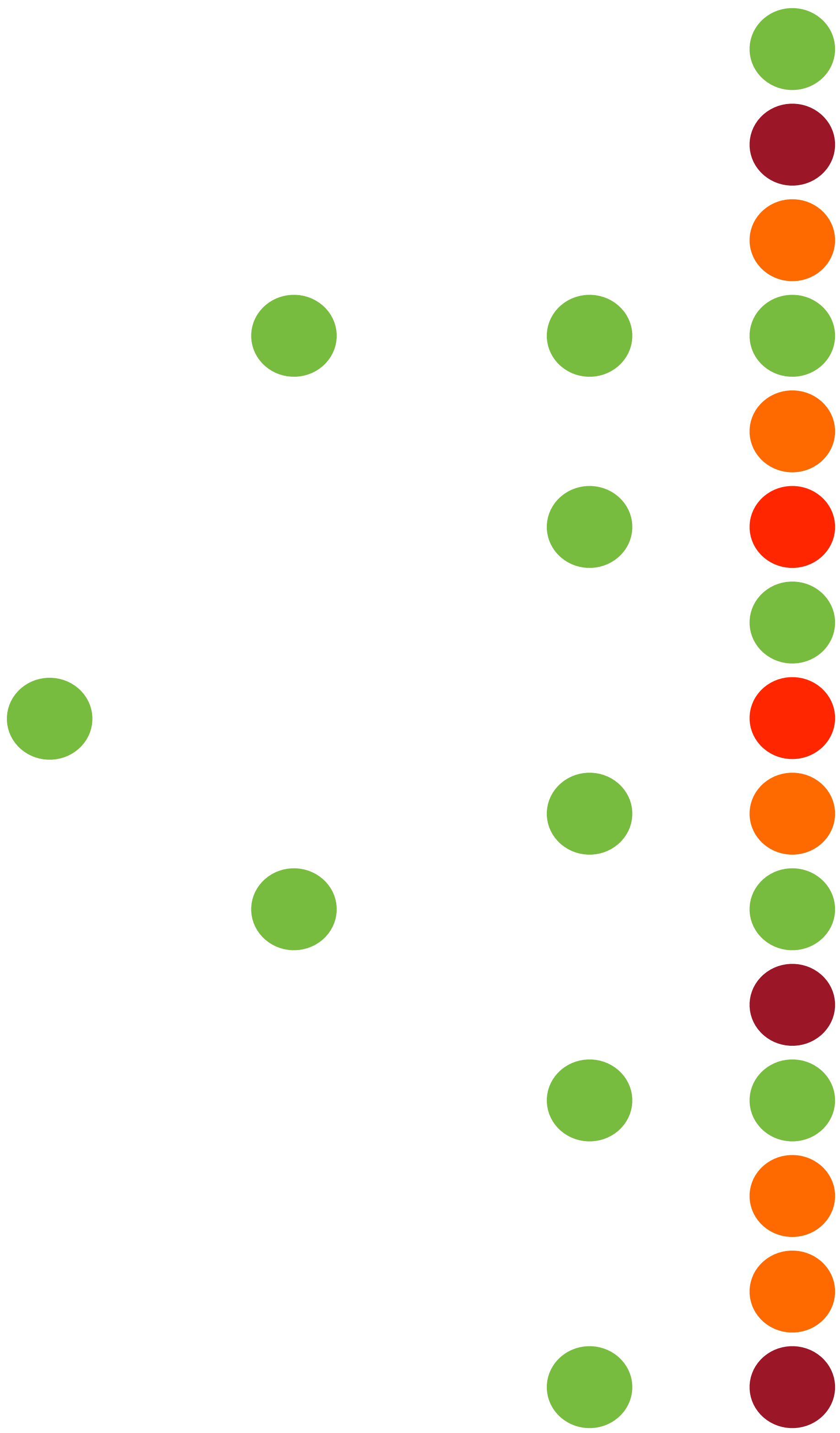
Beam Search



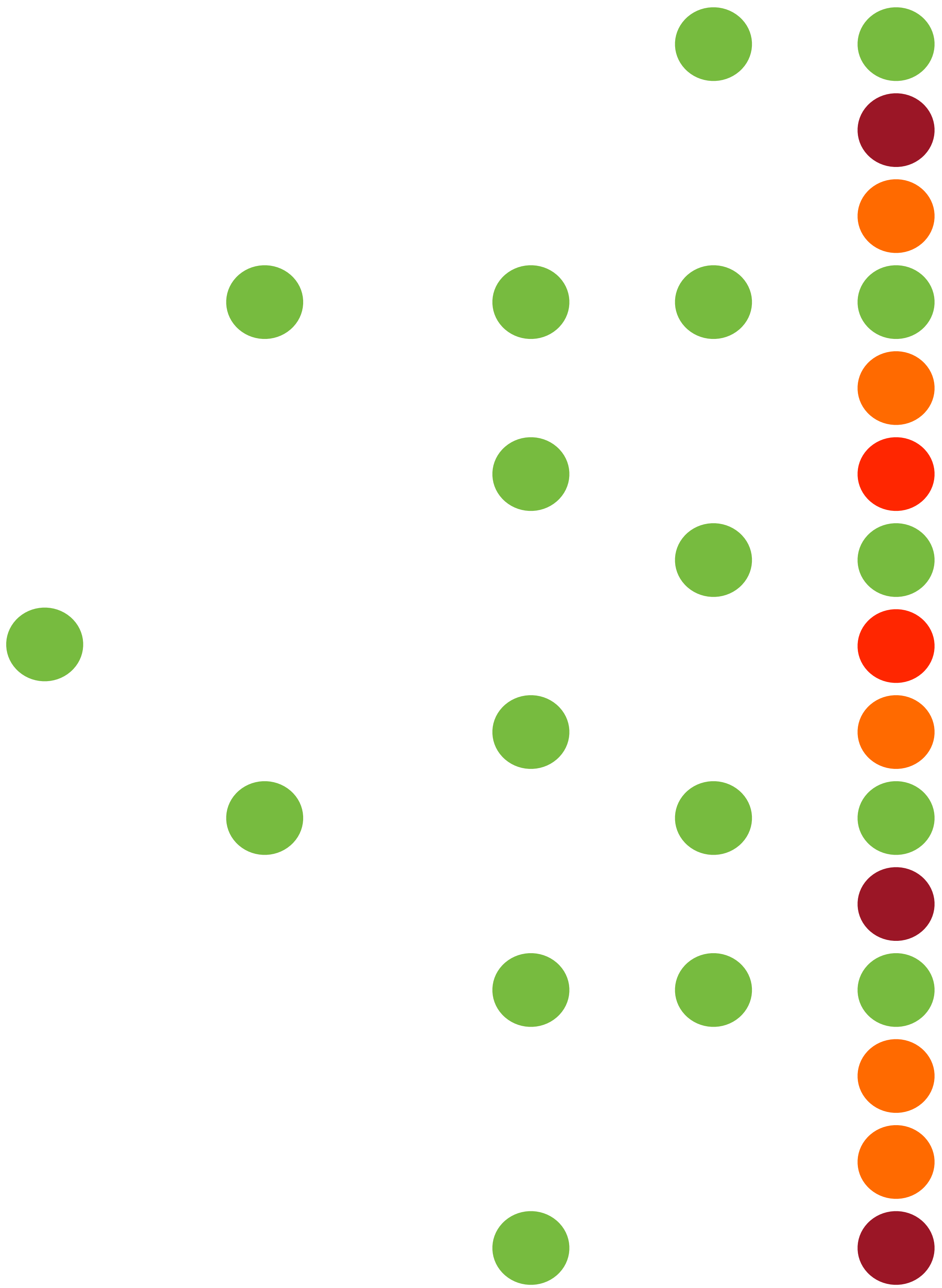
Beam Search



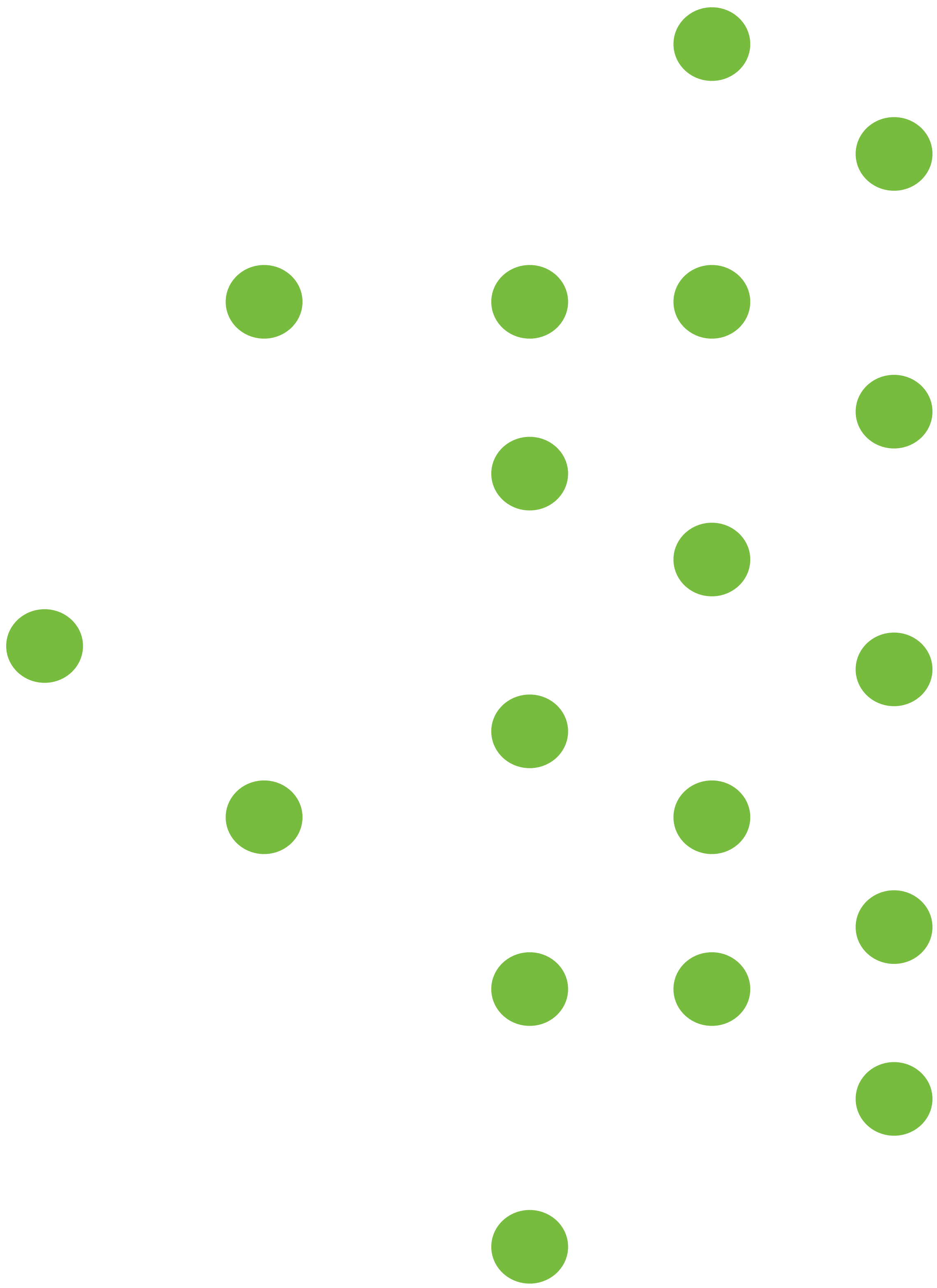
Beam Search



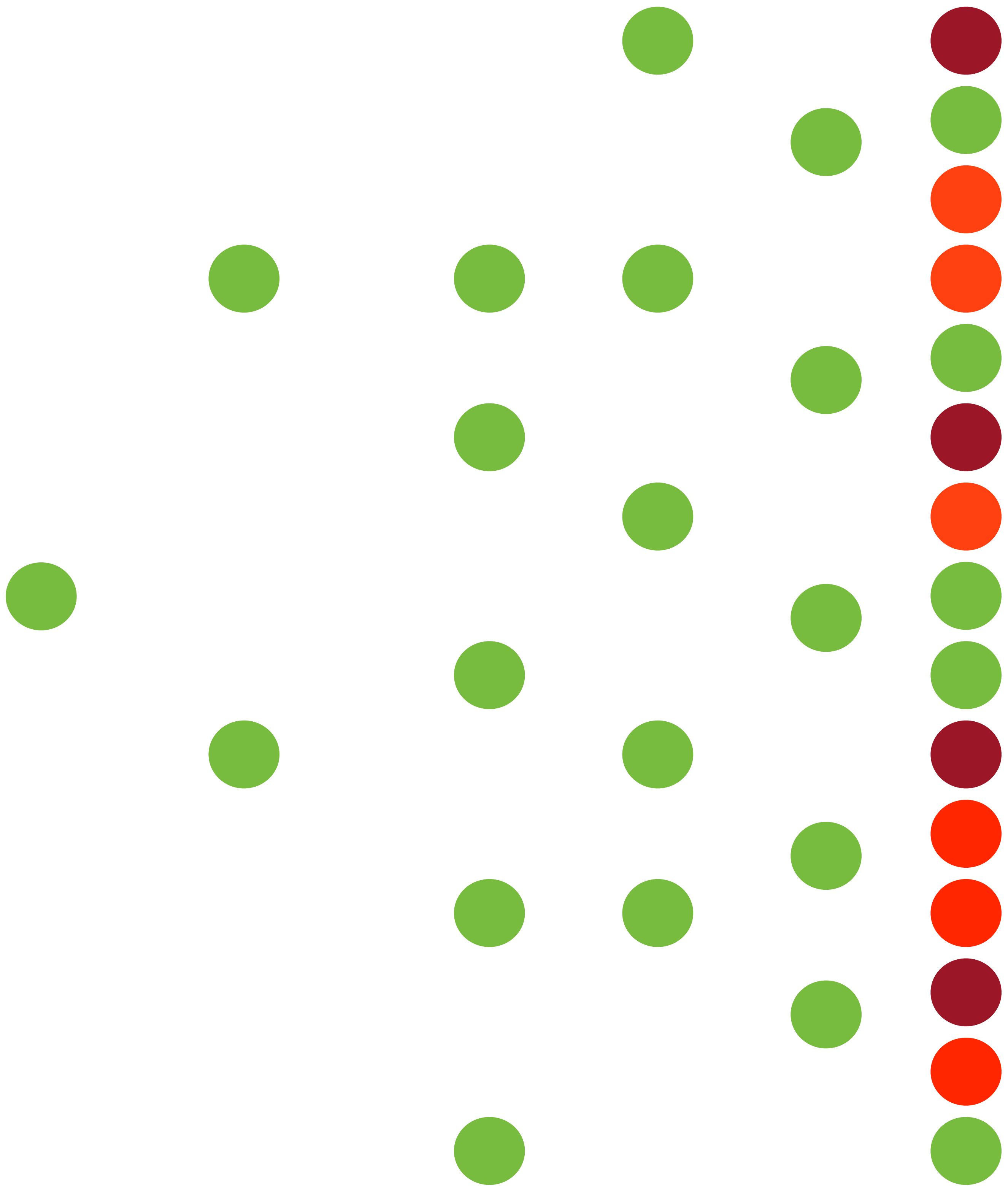
Beam Search



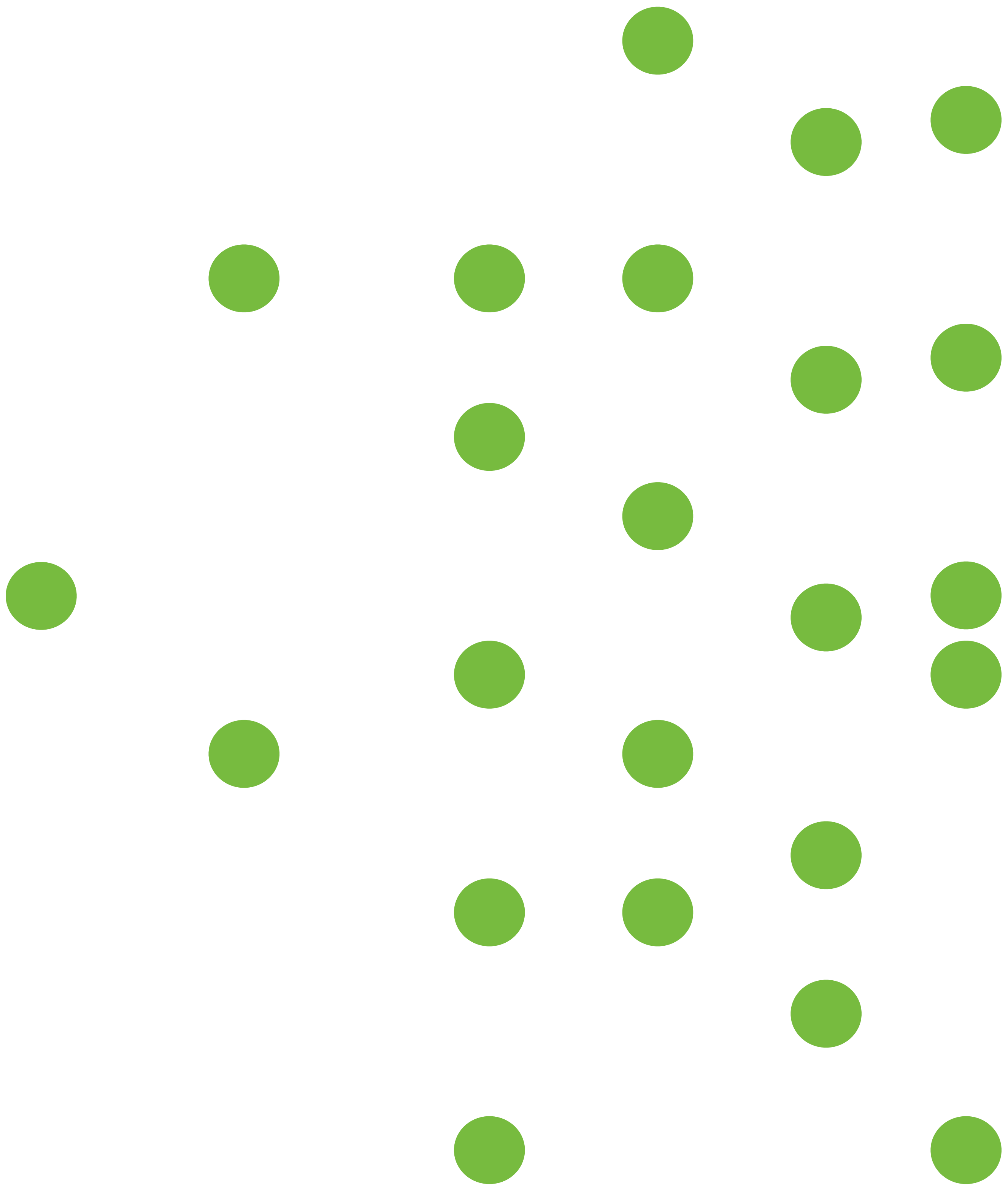
Beam Search



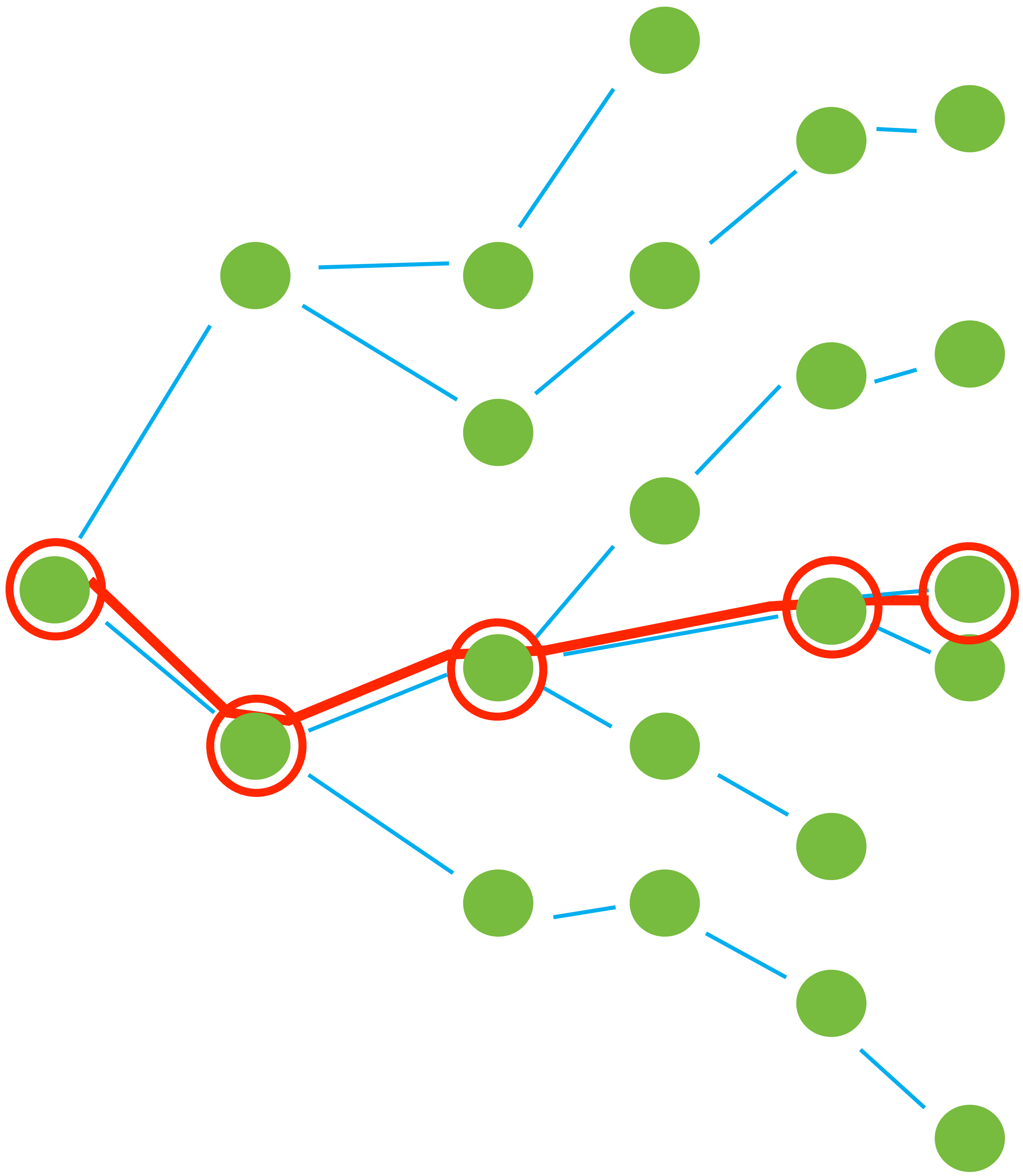
Beam Search



Beam Search



Beam Search



Group Playlisting

- Group playlisting:
  - Radio, Clubs, Offices, Health Clubs, The Web
- Group playlisting challenges:
  - Varying and conflicting music tastes
  - Different levels of assertiveness
- Traditional:
  - Dictator, Compromise
  - Random, Opt-out

Group cost functions

- New cost functions for group playlisting - **social cost function**:
  - Average happiness - group vote of members
  - Maximum happiness - vote of the happiest group member
  - Minimum misery - vote of the least happy

Group costs

	Ben	Paul	Oscar
	2	10	1
	4	3	3
	6	2	7

Group costs

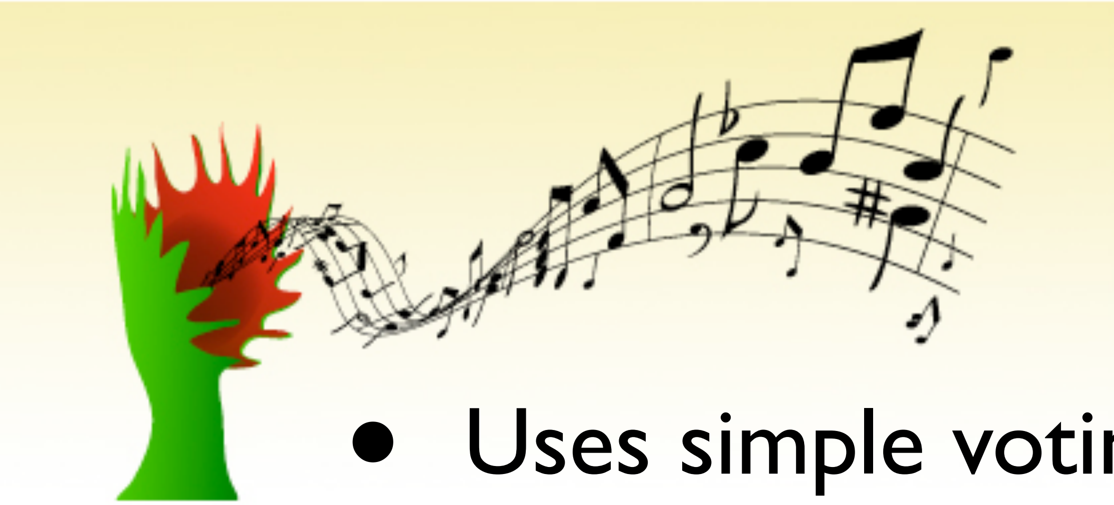
	Ben	Paul	Oscar	Avg
	2	10	1	4.33
	4	3	3	3.33
	6	2	7	**5**

Group costs

	Ben	Paul	Oscar	Avg	Max
	2	10	1	4.33	**10**
	4	3	3	3.33	4
	6	2	7	**5**	6

Group costs

	Ben	Paul	Oscar	Avg	Max	Min
	2	10	1	4.33	**10**	1
	4	3	3	3.33	4	**3**
	6	2	7	**5**	6	2



Flytrap

- Uses simple voting mechanism - ‘average happiness’
  - Each listener agent votes:
    - Artist previously listened == high votes
    - Genre previous listened == positive vote
  - Songs with more votes have higher probability of being played
  - Never play 2 songs by same artist in a row
  - Loose coherence of genre across tracks

Flycasting

1. Translate the request histories of all requesters into ratings for artists.
2. Predict ratings for each artist that a requester has never requested.
3. Determine what artists are the most popular among the listening audience.
4. Determine what artists are similar to the final artist on the playlist.
5. Select a song to play that is performed by an artist that is both popular among the listening requesters and similar to the artist that precedes it.

How to Combine Different Individual Preferences

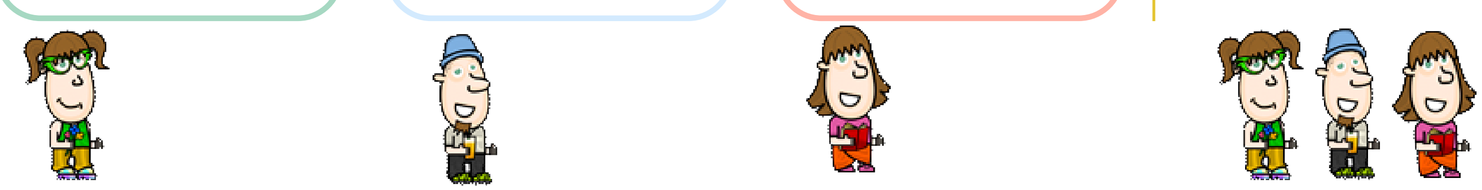
The goal of the Reuse Process is to *combine* different individual preferences into a global **group ranking** of the candidate songs

I Spy (Pulp)

retrieved candidates

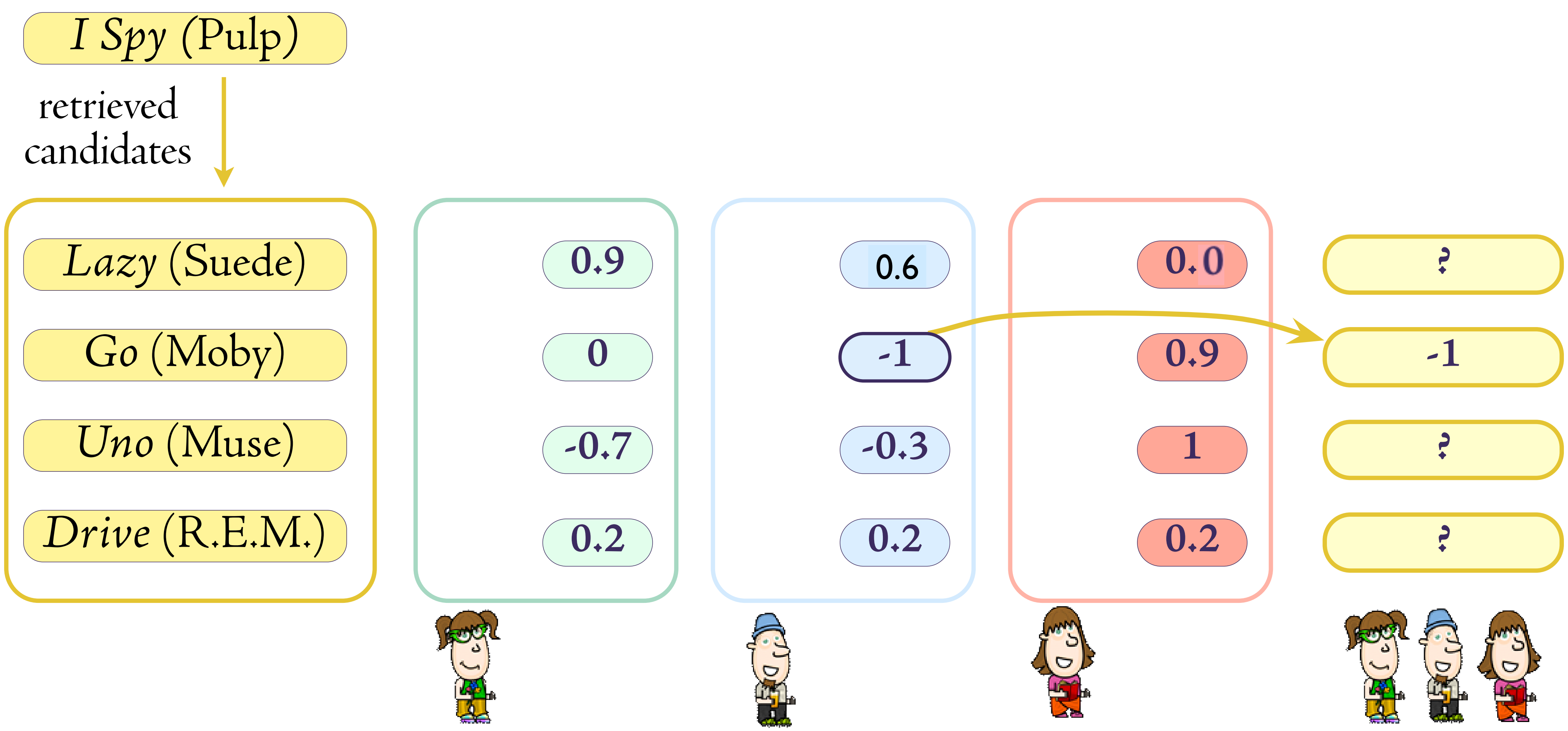
Ex.: three listeners have **diverging** individual preferences over which candidate song to play after *I Spy* (Pulp)

Lazy (Suede)	0.9	0.6	0.0	?
Go (Moby)	0	-1	0.9	?
Uno (Muse)	-0.7	-0.3	1	?
Drive (R.E.M.)	0.2	0.2	0.2	?



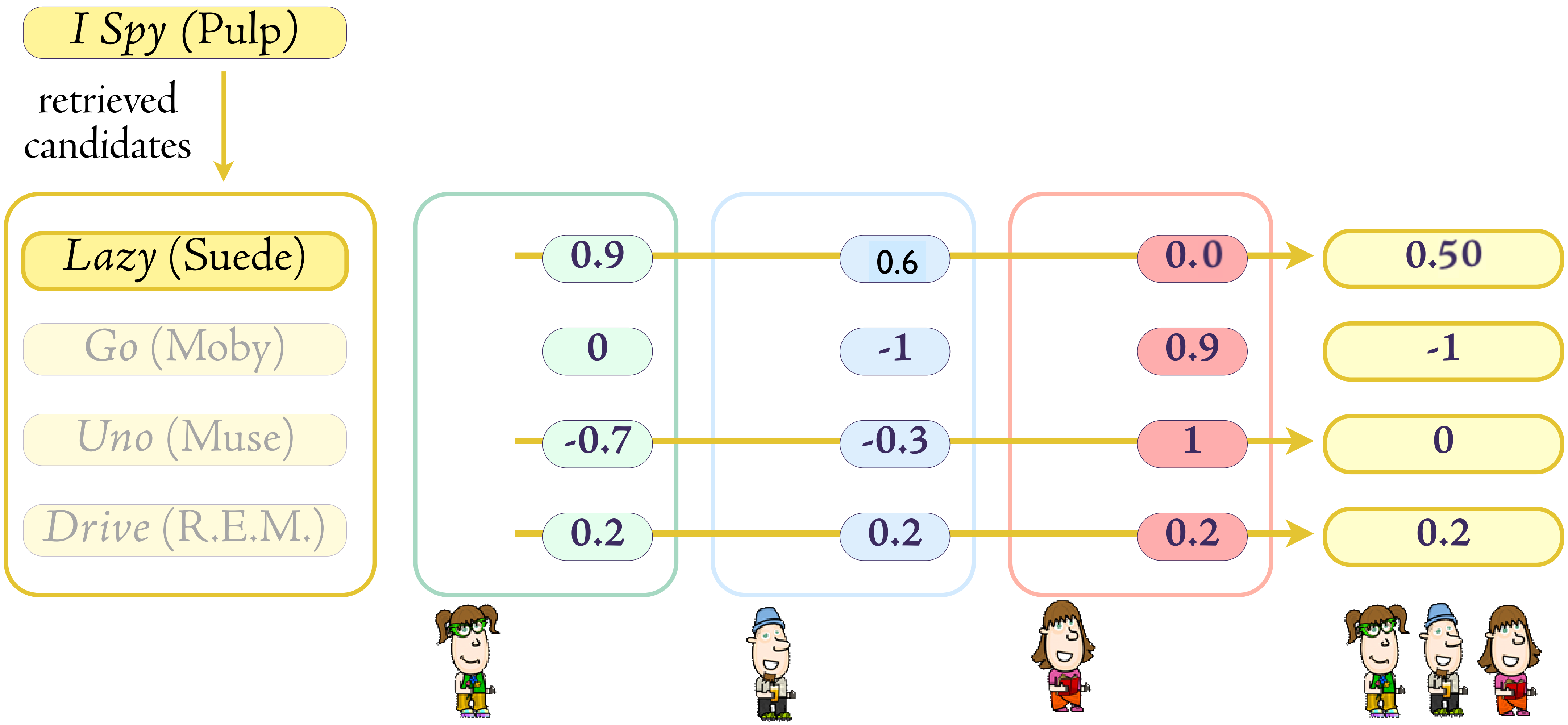
How to Combine Different Individual Preferences

1. To **avoid misery**, any candidate song that is **hated** by some listener automatically gets the lowest group preference degree



How to Combine Different Individual Preferences

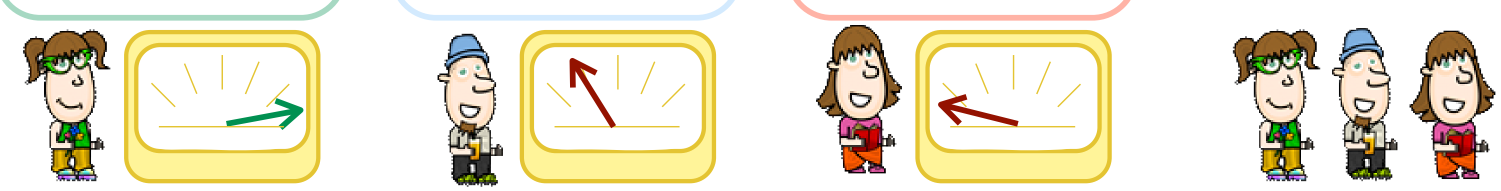
2. To ensure **fairness**, the group preference degree of the remaining candidates equals to the **average** of the individual preferences



How to Combine Different Individual Preferences

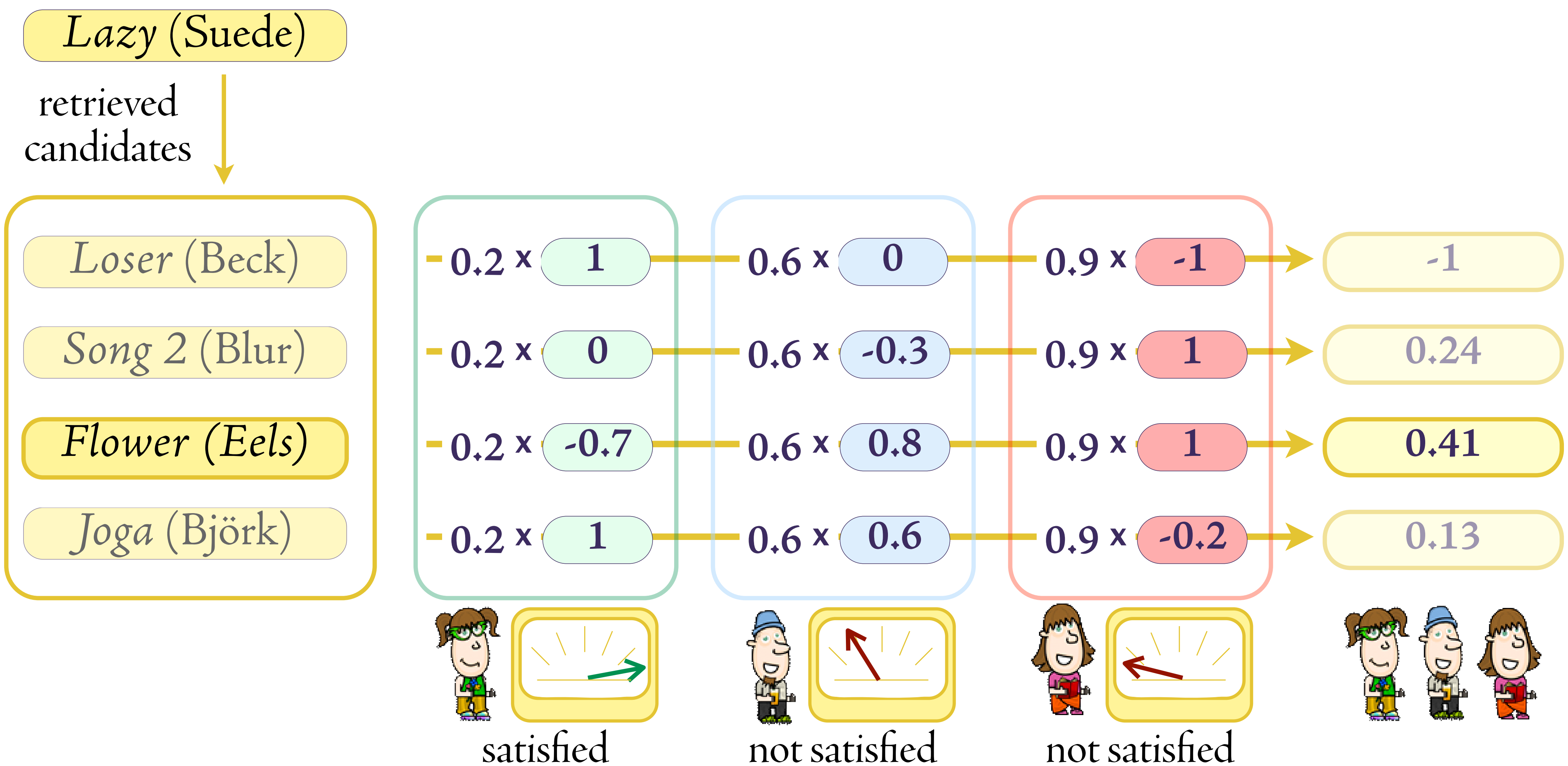
3. To guarantee **individual satisfactions**, listeners whose preferred song was not selected in this turn are to be favoured next

Lazy (Suede)	**0.9**	0.6	0.0	**0.50**
Go (Moby)	0	-1	0.9	-1
Uno (Muse)	-0.7	-0.3	**1**	0
Drive (R.E.M.)	0.2	**0.2**	0.2	0.2



How to Combine Different Individual Preferences

4. The **satisfaction degree** of a listener for previous songs changes her **weight** in the calculation of the average group preference

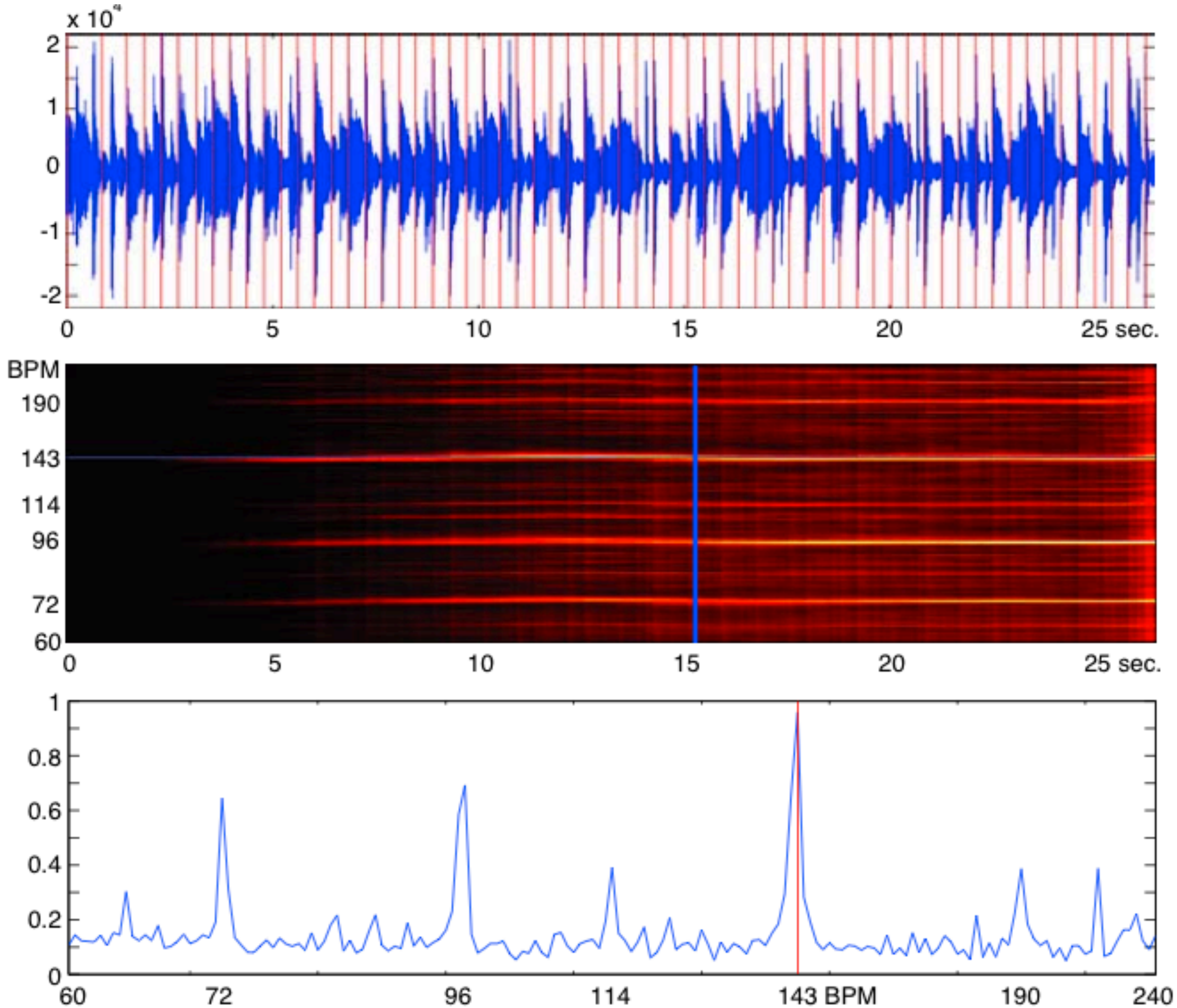


Beat-matching
Crossfading

Beat-matching and crossfading

- Select songs with similar tempos
- Select transition location
  - Similar rhythmic pattern
  - Specific sections (last 30 seconds of song 1 and first 30 seconds of song 2)
- Align their beats over the course of a transition
- Crossfade the volumes

First, find the beats



Time scaling

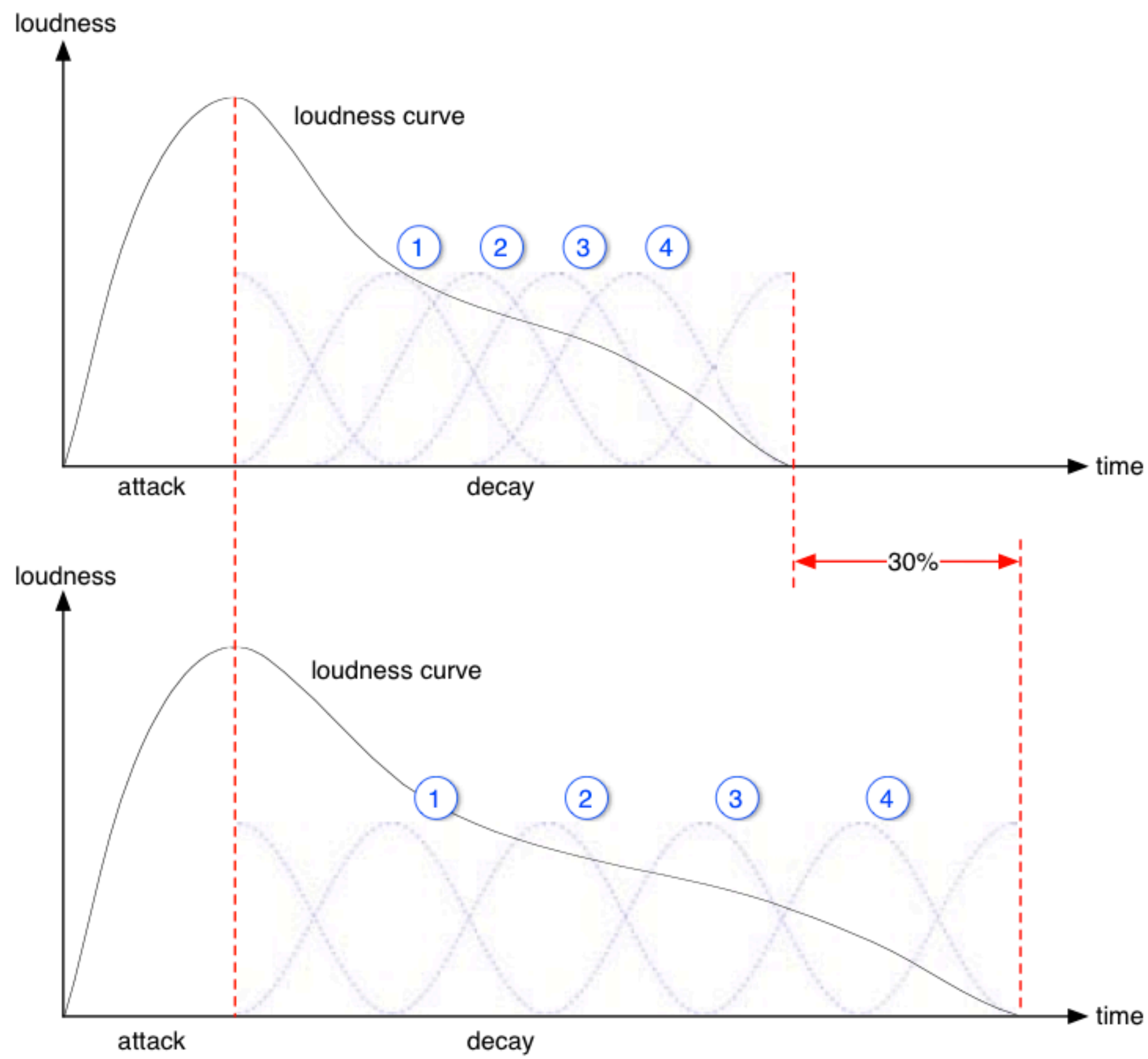
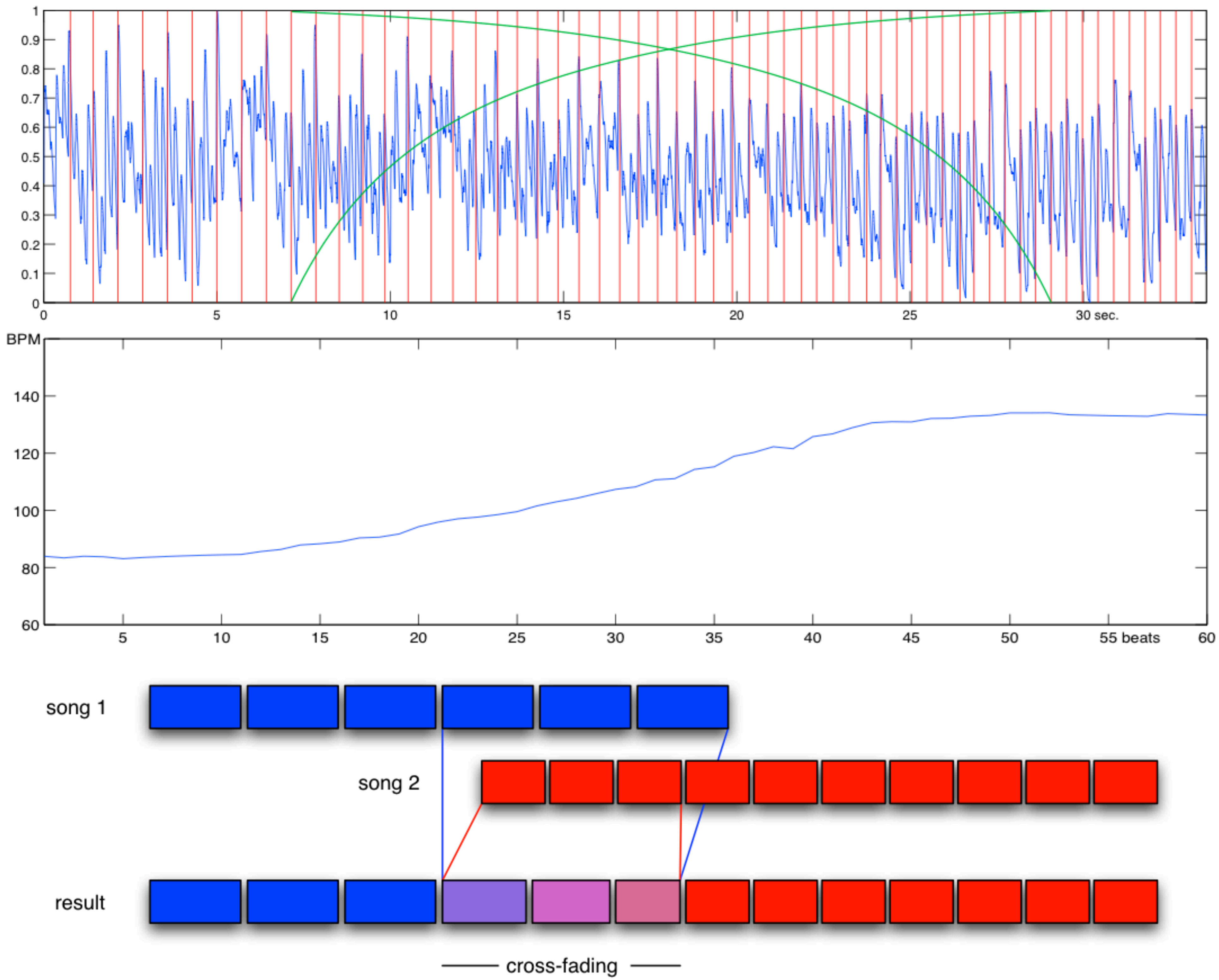


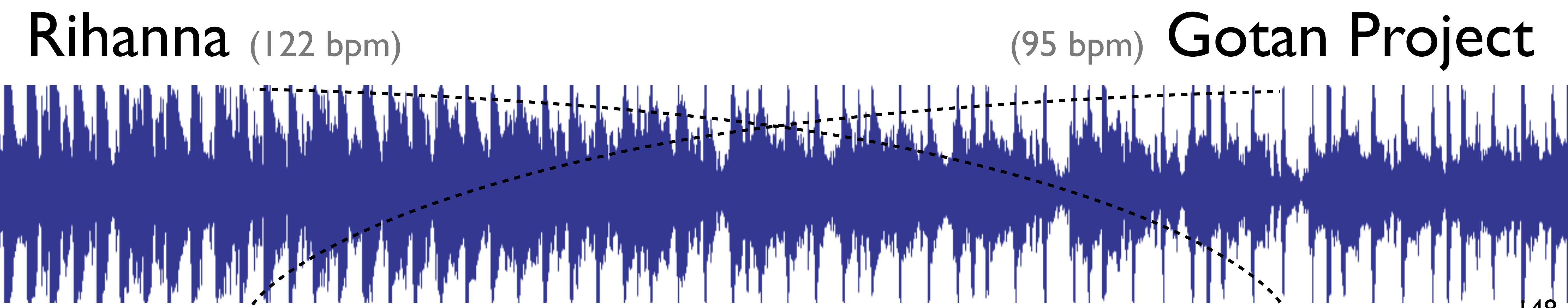
Figure 6-1: Time-scaling example of a typical sound segment. Note that we only process the decay part of the sound. The energy is preserved by overlapping and adding Hanning windows by 50%. In this example we stretch the whole segment [top] by an additional 30% [bottom].

Beat-matching and crossfading



Some Examples

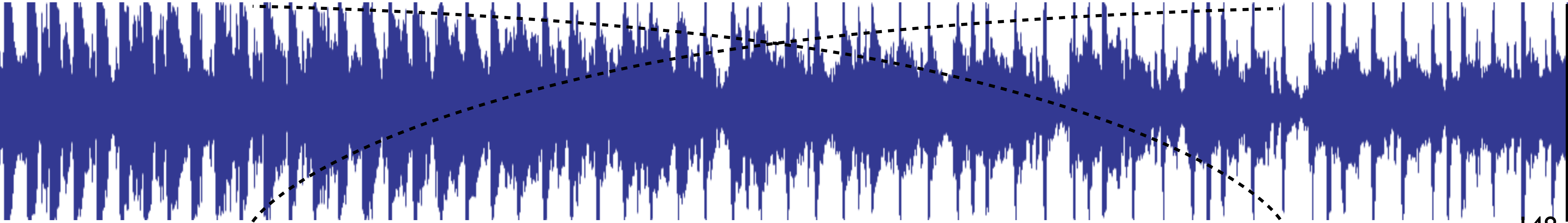
Some Examples



Some Examples

Rihanna (122 bpm)

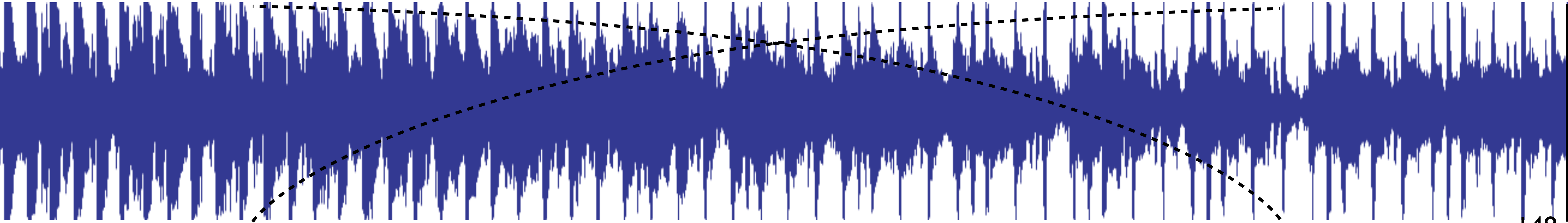
(95 bpm) Gotan Project



Some Examples

Bob Marley to Bob Marley

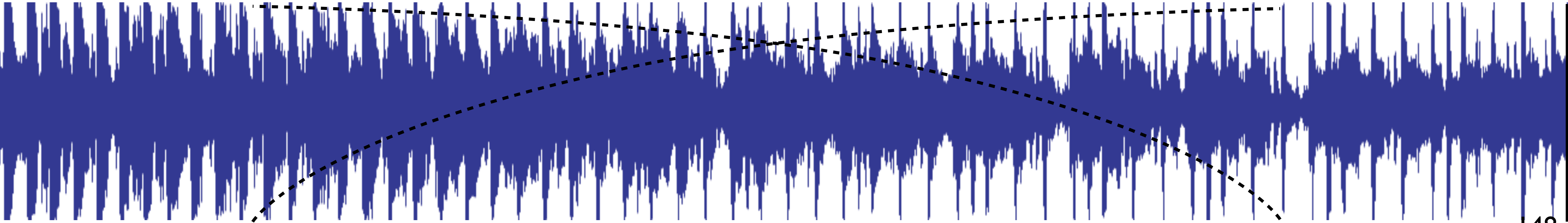
Rihanna (122 bpm) (95 bpm) Gotan Project



Some Examples

Bob Marley to Bob Marley

Rihanna (122 bpm) (95 bpm) Gotan Project

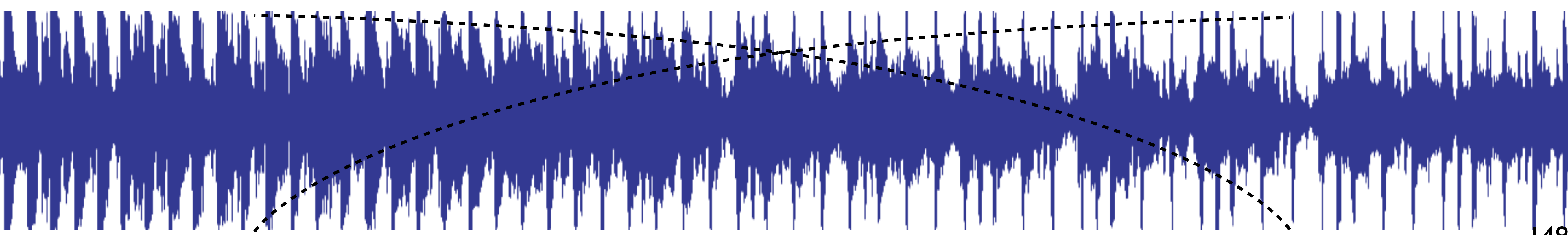


Some Examples

Bob Marley to Bob Marley

Rihanna (122 bpm)

(95 bpm) Gotan Project

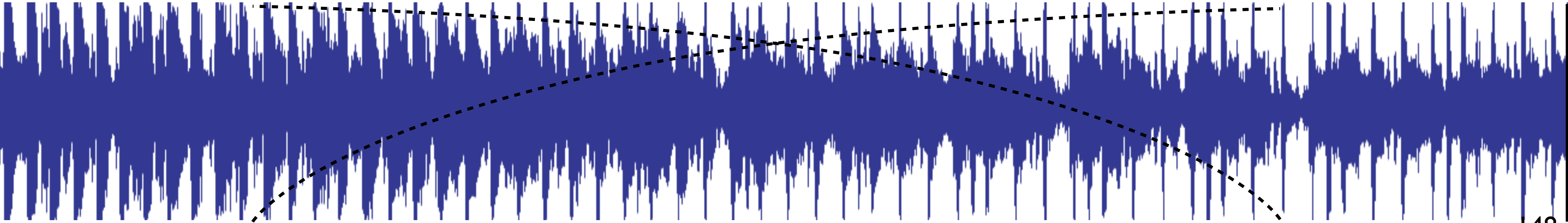


Some Examples

Bob Marley to Bob Marley

Sade to Sting

Rihanna (122 bpm) (95 bpm) Gotan Project

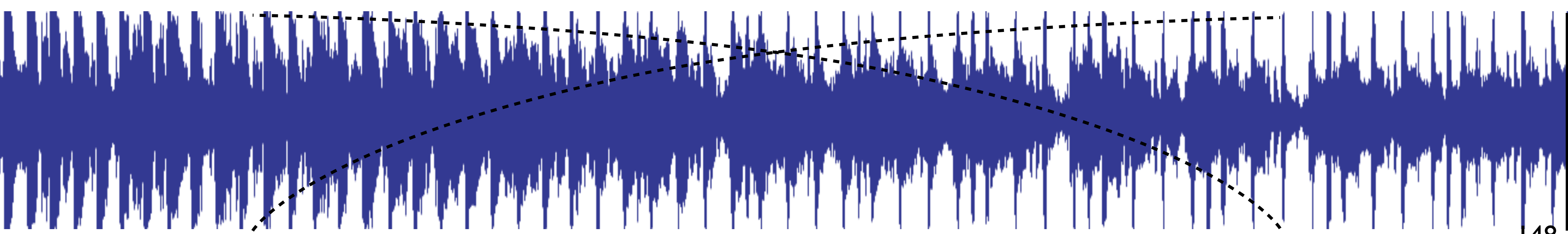


Some Examples

Bob Marley to Bob Marley

Sade to Sting

Rihanna (122 bpm) (95 bpm) Gotan Project

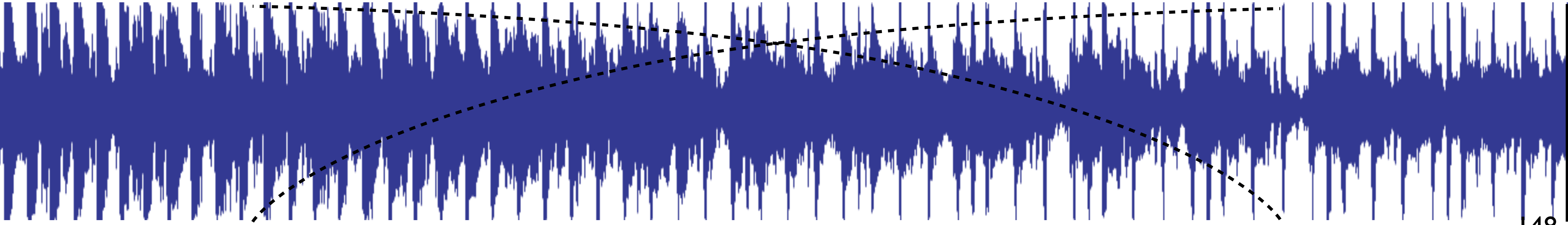


Some Examples

Bob Marley to Bob Marley

Sade to Sting

Rihanna (122 bpm) (95 bpm) Gotan Project



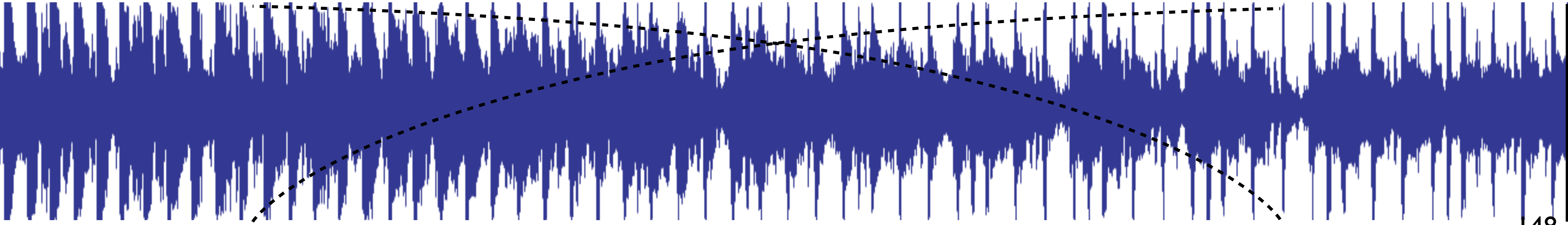
Some Examples

Bob Marley to Bob Marley

Sade to Sting

April March to April March

Rihanna (122 bpm) (95 bpm) Gotan Project

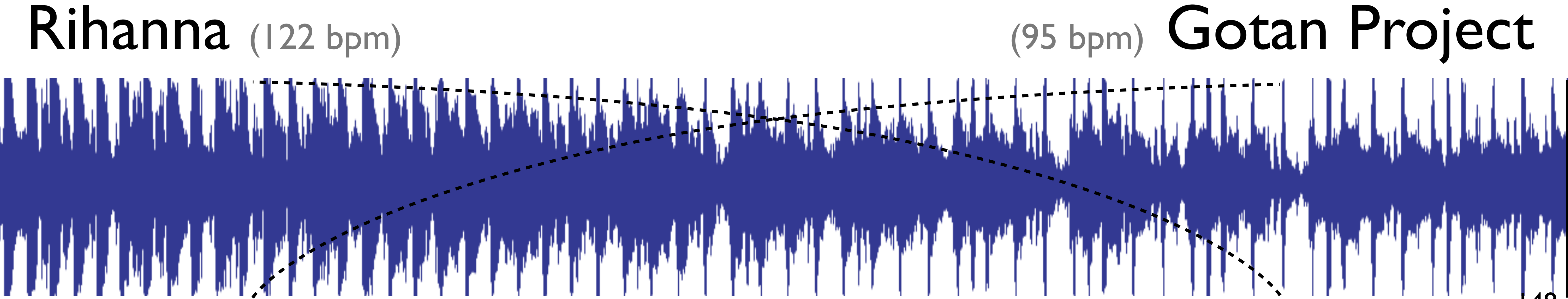


Some Examples

Bob Marley to Bob Marley

Sade to Sting

April March to April March



Evaluating playlists

Subjective Analysis

Direct Listening Tests

hypotheses

1. Playlists compiled by PATS contain more preferred songs than randomly assembled playlists, irrespective of a given context-of-use.
2. Similarly, PATS playlists are rated higher than randomly assembled playlists, irrespective of a given context-of-use.

Direct Listening Tests
hypotheses

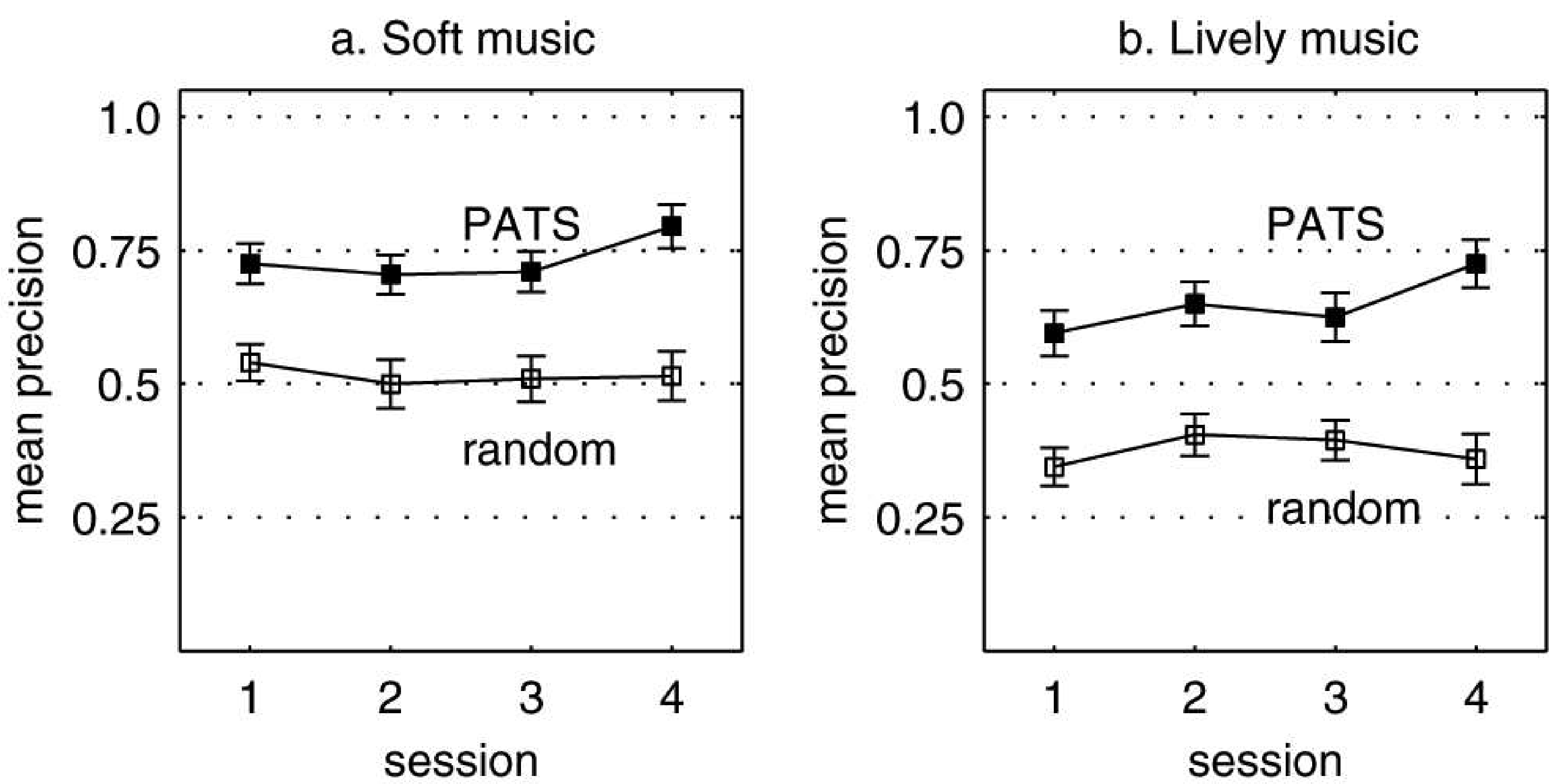
3. Successive playlists compiled by PATS contain an increasing number of preferred songs.
4. Similarly, successive PATS playlists are successively rated higher.
5. Successive playlists compiled by PATS contain more distinct and preferred songs than randomly assembled playlists.

Direct Listening Tests

set-up

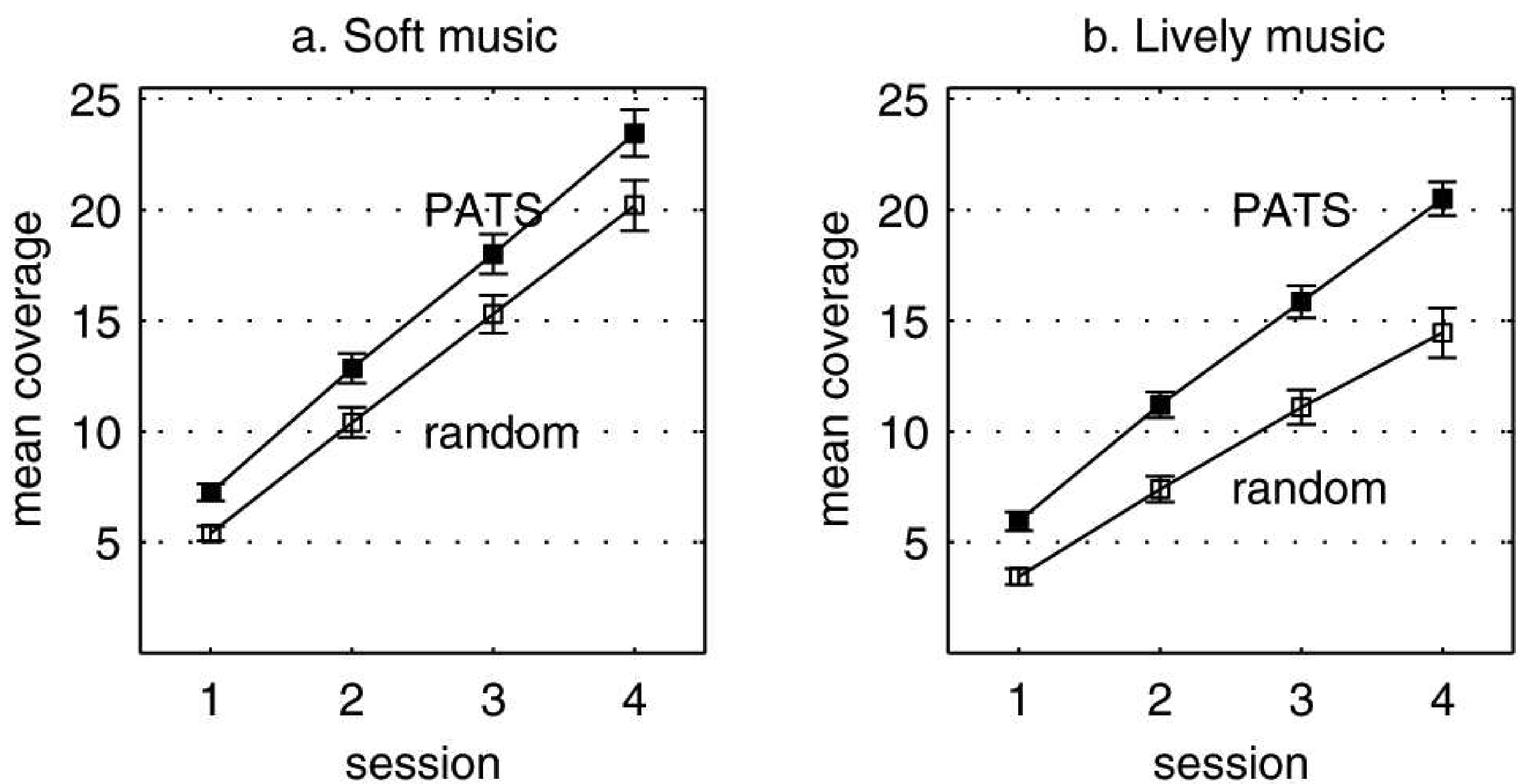
- Three measures: **precision, coverage** and **rating score**
- 20 participants (17m, 3f), 8 sessions over 4 days per participant
    - User selects a song, given a context (4 playlist per context)
    - A PATS playlist and a random playlist are generated (11 songs each, 1 minute excerpts)
    - Judgements expressed per song, ratings per playlist

Direct Listening Tests
results

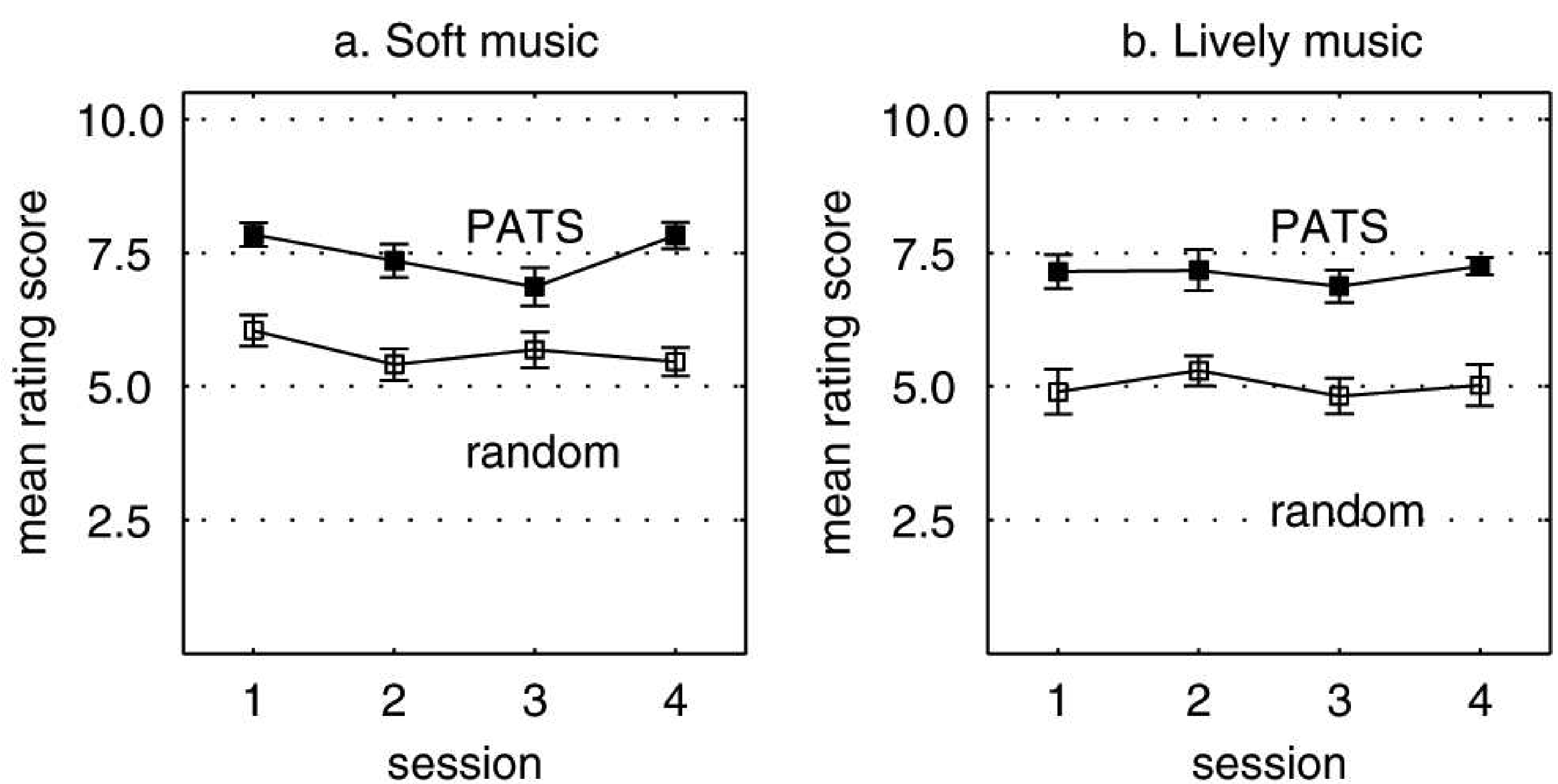


PATS: Realization and User Evaluation of an Automatic Playlist Generator
Steffen Pauws and Berry Eggen

Direct Listening Tests
results



Direct Listening Tests
results



Skip-Based Listening Tests

basics

- Evaluation integrated into system
- Assumptions:

1. A seed song is given
2. A skip button is available and easily accessible to the user
3. A lazy user who is willing to sacrifice quality for time

Skip-Based Listening Tests
use cases

1. The user wants to listen to songs that are similar to the seed song
2. Same as (1) but with a dislike of an arbitrary artist for a subjective reason (eg taste)
3. The user's preference changes over time. Specifically, in a 20 song playlist, the first 5 songs from genre *A*, the middle 10 from either genre *A* or *B*, last 5 songs from genre *B*.

Skip-Based Listening Tests
heuristics

A. *N* nearest neighbors to the seed song are played (*N* = **accepted** + **skipped**). This heuristic is the baseline.

B. The candidate song **closest** to the **last song accepted** by the user is played. This is like (A) except the seed song is always the last song accepted.

C. The candidate song **closest to any** of the accepted songs is played.

D. For each candidate song, let *da* be the distance to the nearest accepted, and let *ds* be the distance to the nearest skipped. If *da* < *ds*, then add the candidate to the set *S*. From *S* play the song with smallest *da*. If *S* is empty, then play the candidate song which has the best (i.e. the lowest) *da*/*ds* ratio.

Dynamic Playlist Generation Based on Skipping Behavior
Elias Pampalk and T. Pohle and G. Widmer

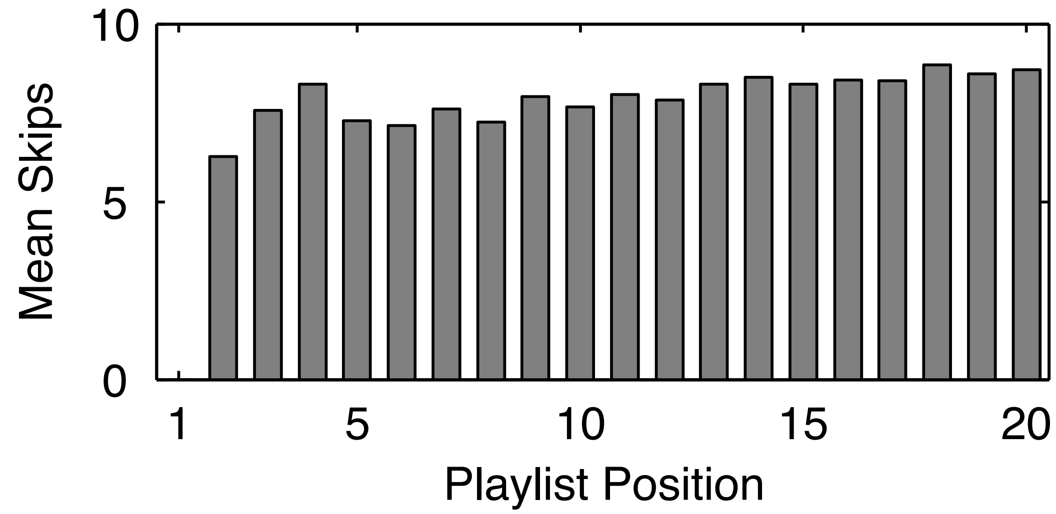
Skip-Based Listening Tests

skips in UC1

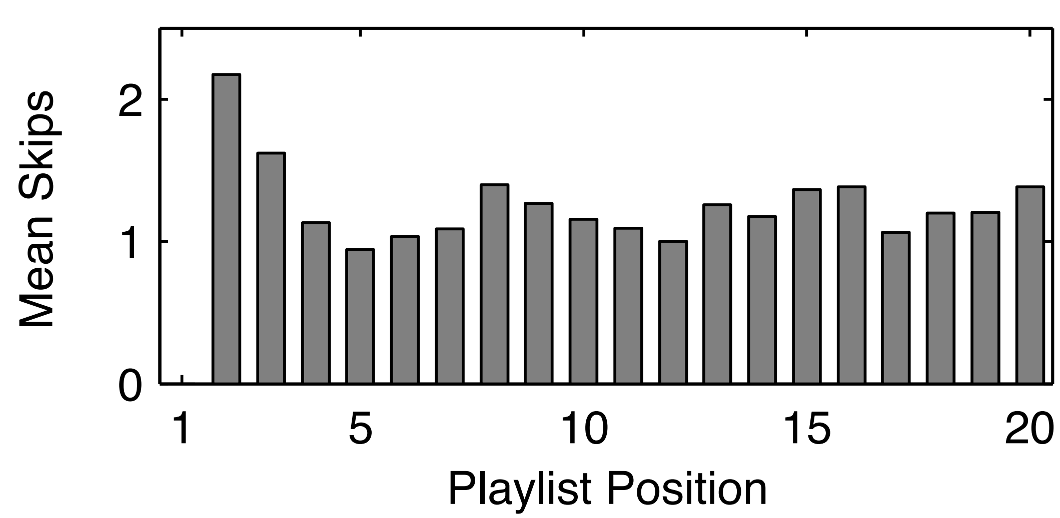
			Artists/Genre		Tracks/Genre	
Genres	Artists	Tracks	Min	Max	Min	Max
22	103	2522	3	6	45	259

Skip-Based Listening Tests

skips in UC1



(a) Heuristic A



(b) Heuristic D

Skip-Based Listening Tests
UC1 and UC2 skips

	Heuristic	Min	Median	Mean	Max
UC-1	A	0	37.0	133.0	2053
	B	0	30.0	164.4	2152
	C	0	14.0	91.0	1298
	D	0	11.0	23.9	425
UC-2	A	0	52.0	174.0	2230
	B	0	36.0	241.1	2502
	C	0	17.0	116.9	1661
	D	0	15.0	32.9	453

Skip-Based Listening Tests

UC1 and UC2 skips

	Heuristic	Min	Median	Mean	Max
UC-1	A	0	37.0	133.0	2053
	B	0	30.0	164.4	2152
	C	0	14.0	91.0	1298
	D	0	11.0	23.9	425
UC-2	A	0	52.0	174.0	2230
	B	0	36.0	241.1	2502
	C	0	17.0	116.9	1661
	D	0	15.0	32.9	453

Skip-Based Listening Tests

UC1 and UC2 skips

	Heuristic	Min	Median	Mean	Max
UC-1	A	0	37.0	133.0	2053
	B	0	30.0	164.4	2152
	C	0	14.0	91.0	1298
	D	0	11.0	23.9	425
UC-2	A	0	52.0	174.0	2230
	B	0	36.0	241.1	2502
	C	0	17.0	116.9	1661
	D	0	15.0	32.9	453

Skip-Based Listening Tests

UC3 skips

Start	Goto	Heuristic A Median	Heuristic A Mean	Heuristic B Median	Heuristic B Mean	Heuristic C Median	Heuristic C Mean	Heuristic D Median	Heuristic D Mean
Euro-Dance	Trance	69.0	171.4	36.0	64.9	41.0	69.0	20.0	28.3
Trance	Euro-Dance	66.0	149.1	24.0	79.1	6.5	44.4	4.5	8.8
German Hip Hop	Hard Core Rap	33.0	61.9	32.0	45.6	31.0	40.7	23.0	28.1
Hard Core Rap	German Hip Hop	21.5	32.2	18.0	51.9	16.0	24.2	14.0	16.1
Heavy Metal/Thrash	Death Metal	98.5	146.4	54.0	92.5	58.0	61.1	28.0	28.4
Death Metal	Heavy Metal/Thrash	14.0	69.2	16.0	53.7	3.0	55.5	3.0	25.7
Bossa Nova	Jazz Guitar	68.5	228.1	32.0	118.7	54.0	61.1	22.0	21.3
Jazz Guitar	Bossa Nova	21.0	26.7	22.0	21.5	9.0	10.5	6.0	6.2
Jazz Guitar	Jazz	116.0	111.3	53.0	75.7	45.0	74.0	18.5	27.3
Jazz	Jazz Guitar	512.5	717.0	1286.0	1279.5	311.0	310.8	29.0	41.3
A Cappella	Death Metal	1235.0	1230.5	1523.0	1509.9	684.0	676.5	271.0	297
Death Metal	A Cappella	1688.0	1647.2	1696.0	1653.9	1186.0	1187.3	350.0	309.2

Dynamic Playlist Generation Based on Skipping Behavior
Elias Pampalk and T. Pohle and G. Widmer

Skip-Based Listening Tests

UC3 skips

Start	Goto	Heuristic A Median	Heuristic A Mean	Heuristic B Median	Heuristic B Mean	Heuristic C Median	Heuristic C Mean	Heuristic D Median	Heuristic D Mean
Euro-Dance	Trance	69.0	171.4	36.0	64.9	41.0	69.0	20.0	28.3
Trance	Euro-Dance	66.0	149.1	24.0	79.1	6.5	44.4	4.5	8.8
German Hip Hop	Hard Core Rap	33.0	61.9	32.0	45.6	31.0	40.7	23.0	28.1
Hard Core Rap	German Hip Hop	21.5	32.2	18.0	51.9	16.0	24.2	14.0	16.1
Heavy Metal/Thrash	Death Metal	98.5	146.4	54.0	92.5	58.0	61.1	28.0	28.4
Death Metal	Heavy Metal/Thrash	14.0	69.2	16.0	53.7	3.0	55.5	3.0	25.7
Bossa Nova	Jazz Guitar	68.5	228.1	32.0	118.7	54.0	61.1	22.0	21.3
Jazz Guitar	Bossa Nova	21.0	26.7	22.0	21.5	9.0	10.5	6.0	6.2
Jazz Guitar	Jazz	116.0	111.3	53.0	75.7	45.0	74.0	18.5	27.3
Jazz	Jazz Guitar	512.5	717.0	1286.0	1279.5	311.0	310.8	29.0	41.3
A Cappella	Death Metal	1235.0	1230.5	1523.0	1509.9	684.0	676.5	271.0	297
Death Metal	A Cappella	1688.0	1647.2	1696.0	1653.9	1186.0	1187.3	350.0	309.2

Dynamic Playlist Generation Based on Skipping Behavior
Elias Pampalk and T. Pohle and G. Widmer

Skip-Based Listening Tests

UC3 skips

Start	Goto	Heuristic A Median	Heuristic A Mean	Heuristic B Median	Heuristic B Mean	Heuristic C Median	Heuristic C Mean	Heuristic D Median	Heuristic D Mean
Euro-Dance	Trance	69.0	171.4	36.0	64.9	41.0	69.0	20.0	28.3
Trance	Euro-Dance	66.0	149.1	24.0	79.1	6.5	44.4	4.5	8.8
German Hip Hop	Hard Core Rap	33.0	61.9	32.0	45.6	31.0	40.7	23.0	28.1
Hard Core Rap	German Hip Hop	21.5	32.2	18.0	51.9	16.0	24.2	14.0	16.1
Heavy Metal/Thrash	Death Metal	98.5	146.4	54.0	92.5	58.0	61.1	28.0	28.4
Death Metal	Heavy Metal/Thrash	14.0	69.2	16.0	53.7	3.0	55.5	3.0	25.7
Bossa Nova	Jazz Guitar	68.5	228.1	32.0	118.7	54.0	61.1	22.0	21.3
Jazz Guitar	Bossa Nova	21.0	26.7	22.0	21.5	9.0	10.5	6.0	6.2
Jazz Guitar	Jazz	116.0	111.3	53.0	75.7	45.0	74.0	18.5	27.3
Jazz	Jazz Guitar	512.5	717.0	1286.0	1279.5	311.0	310.8	29.0	41.3
A Cappella	Death Metal	1235.0	1230.5	1523.0	1509.9	684.0	676.5	271.0	297
Death Metal	A Cappella	1688.0	1647.2	1696.0	1653.9	1186.0	1187.3	350.0	309.2

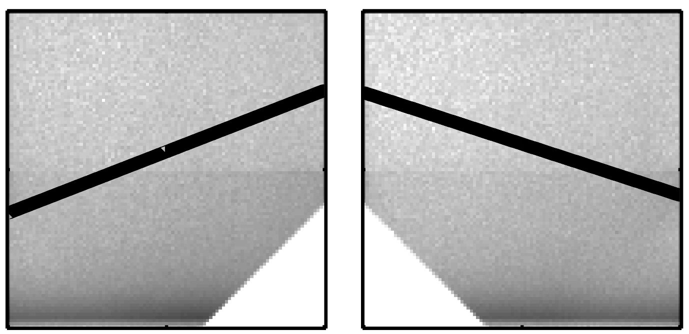
Dynamic Playlist Generation Based on Skipping Behavior
Elias Pampalk and T. Pohle and G. Widmer

Dynamic Heuristics

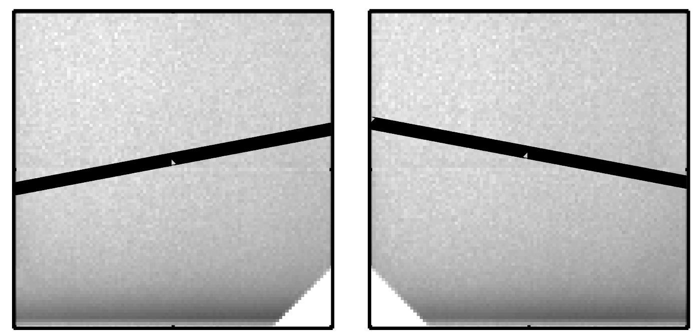
- Last.fm Radio logs are used to analyze and evaluate several heuristics for dynamic playlists
- This is done through the treatment of playlists as fuzzy sets
- Work shows that one heuristic work best given **inconsistent rejects** while another performs best given **inconsistent accepts** and third performs equally in either environment.

Evaluating and Analysing Dynamic Playlist Generation Heuristics Using Radio Logs and Fuzzy Set Theory
Bosteels, Klaas and Pampalk, Elias and Kerre, Etienne E.

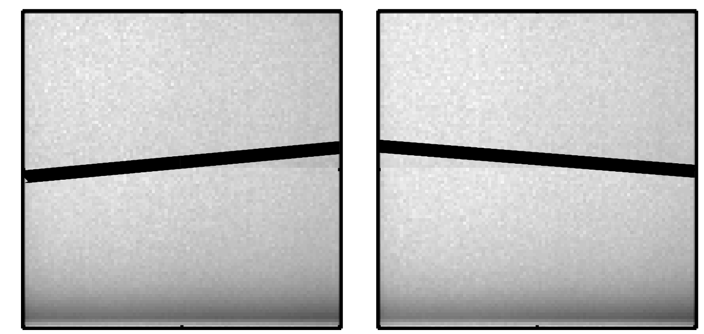
Dynamic Heuristics



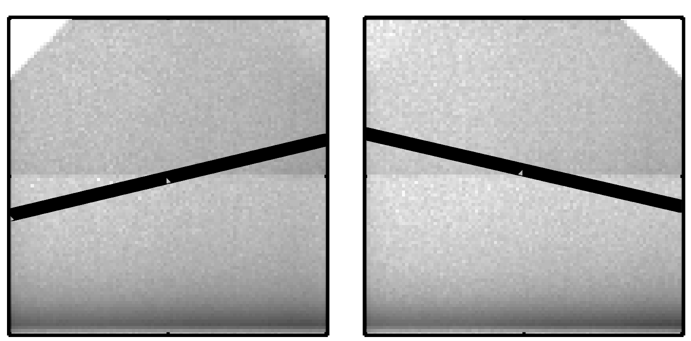
(a) dataset 1



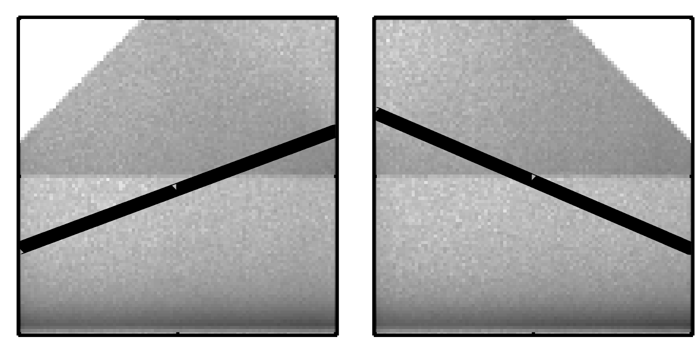
(b) dataset 3



(c) dataset 5

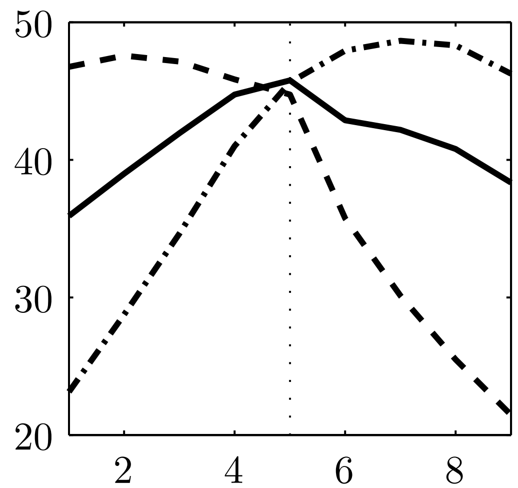


(d) dataset 7

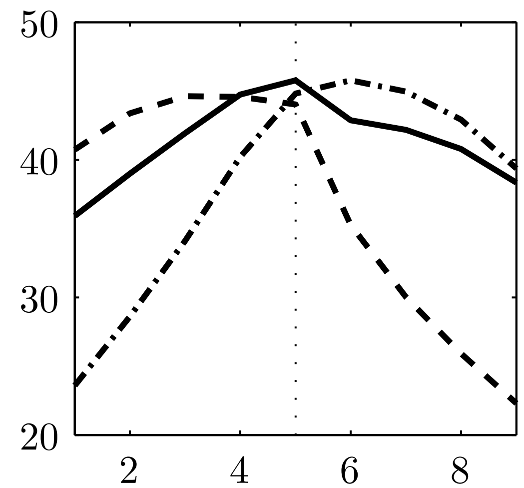


(e) dataset 9

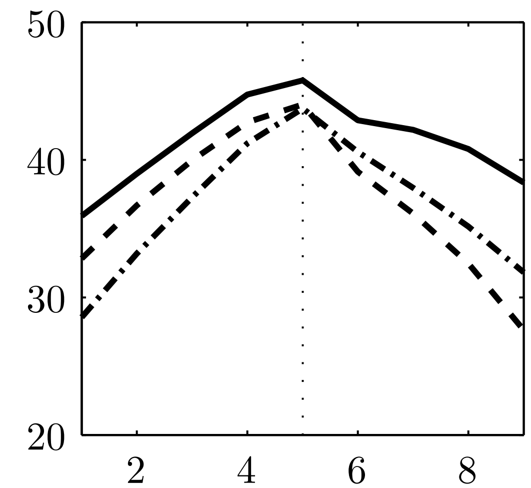
Dynamic Heuristics



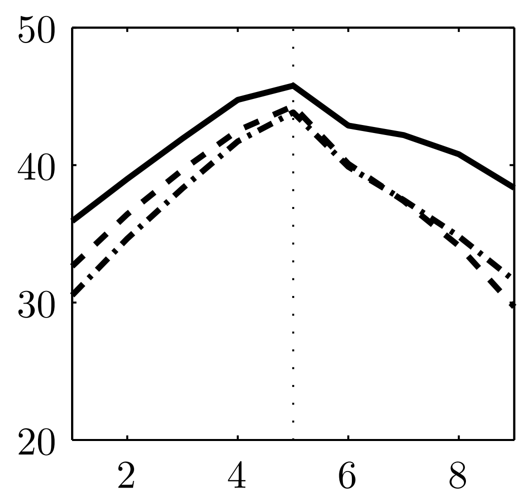
(a) $I_{S_{\mathrm{M}}}$



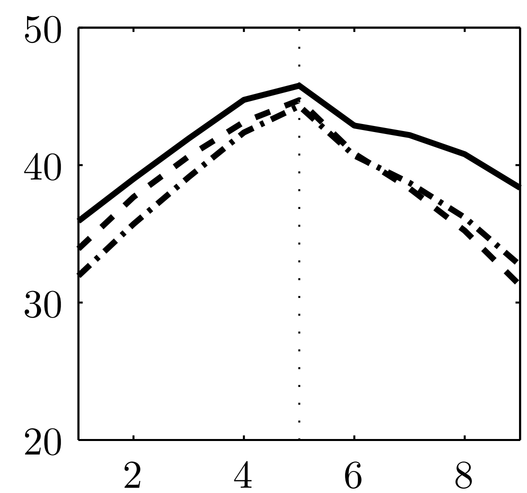
(b) $I_{S_{\mathrm{P}}}$



(c) $I_{S_{\mathrm{L}}} = I_{T_{\mathrm{L}}}$



(d) $I_{T_{\mathrm{P}}}$



(e) $I_{T_{\mathrm{M}}}$

Evaluating and Analysing Dynamic Playlist Generation Heuristics Using Radio Logs and Fuzzy Set Theory
Bosteels, Klaas and Pampalk, Elias and Kerre, Etienne E.

objective analysis

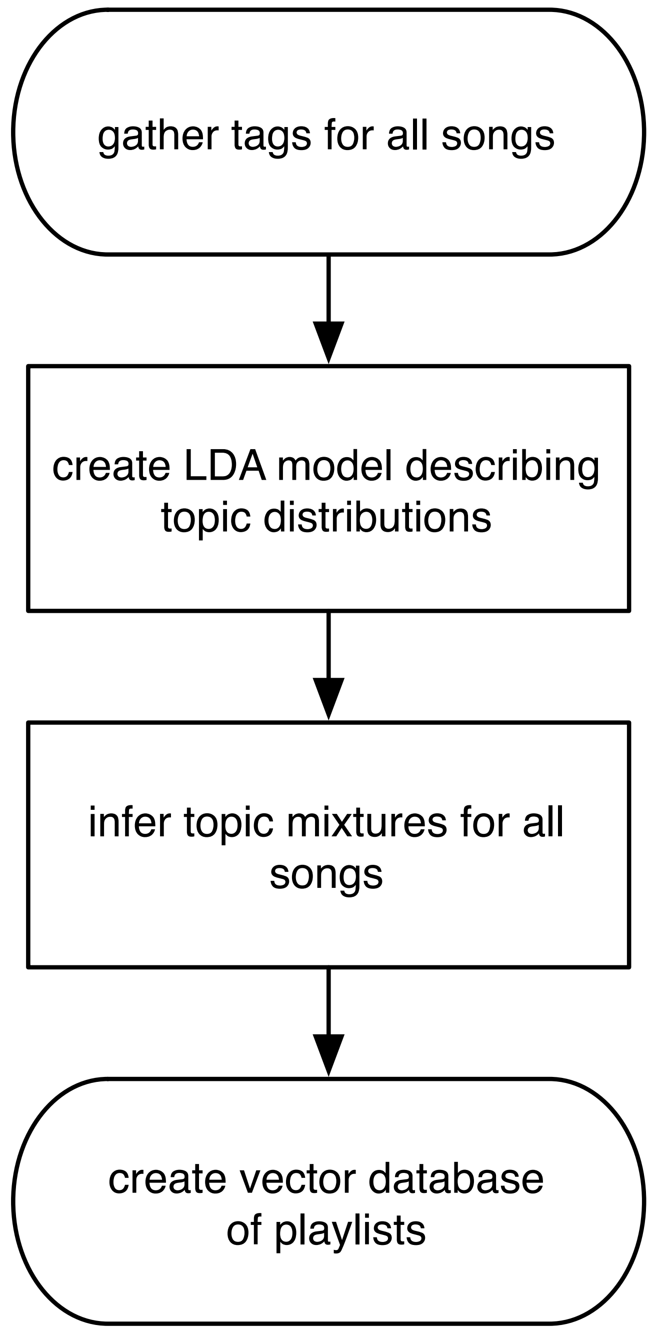
Measuring Distance

We can measure the distance between sequences of **tracks** using the same methods we can use to measure the distance between **frames** within tracks.

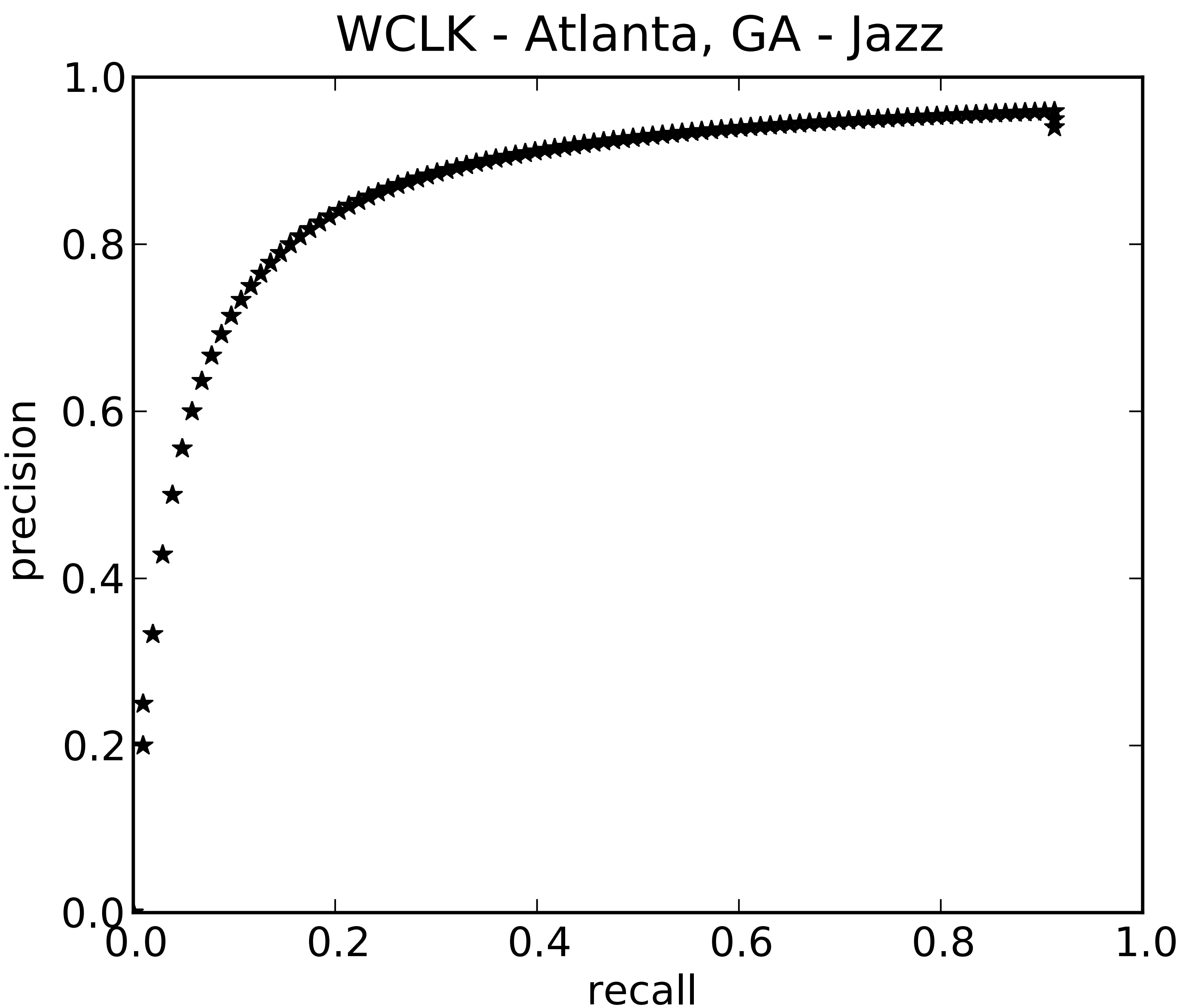
Measuring Distance

- Topic Modeled Tag Clouds used as a song-level feature
- Sequences of these low dimensional features can then be compared
- The fitness of this pseudo-metric space is examined through patterns in radio playlist logs

Measuring Distance



Measuring Distance



An evaluation of various playlisting services

Radio Paradise

RP is a blend of many styles and genres of music, carefully selected and mixed by two real human beings. You'll hear modern and classic rock, world music, electronica, even a bit of classical and jazz. What you won't hear are random computer-generated playlists or mind-numbing commercials.

Our specialty is taking a diverse assortment of songs and making them flow together in a way that makes sense harmonically, rhythmically, and lyrically — an art that, to us, is the very essence of radio

- Listener supported, Internet streaming radio
- 75,000 registered listeners
- Real, radio DJs

radioparadise.com

Playlist Turing Test

The Playlist Survey

In this survey we are looking at the quality of playlists generated by human experts, computer algorithms and random number generators. You will be presented with 12 playlists. For each playlist, we ask you to rate the overall quality of the playlist as a whole and to predict whether the playlist was generated by a professional DJ, a computer algorithm or was created at random. At the end of the survey you'll be shown how accurate your predictions were. More info about this survey can be found at The Playlist Survey

What is your age?

What is your gender? not specified

How often do you listen to Radio Paradise? not specified

Take the Survey

Expert playlist data graciously provided by Radio Paradise

RADIO PARADISE Listener Supported

380 respondents

Radio Paradise Dataset

Playlists collected from Jan 2007 to July 2008

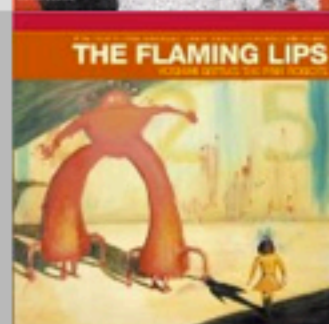
Playlists	45,283
Tracks	6,325
Albums	4,094
Artists	1971
Average Length	4.3

Playlist Turing Test

The Playlist Survey

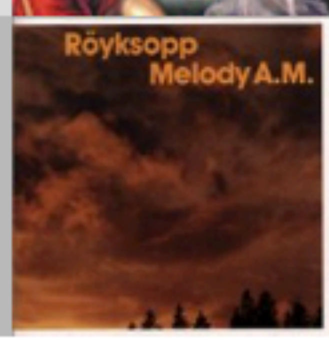
12 Playlists remaining

Fresh Feeling
Eels
Play on Spotify
Play in youtube



Are You A Hypnotist??
Flaming Lips
Play on Spotify
Play in youtube

Video
Aimee Mann
Play in youtube



Poor Leno
R�yksopp
Play on Spotify
Play in youtube

Playlist rating? not specified   Playlist type? not specified   Submit

The Survey

Survey Ratings for guesses

Playlist guess	Playlist Rating	Counts
Human Expert	3.33	368
Algorithm	2.76	373
Random	2.08	343

The Survey

Survey Ratings for truth

Playlist type	Playlist Rating	Counts
Human Expert	2.49	400
Algorithm	2.63	403
Random	2.64	386

The Survey

Confusion Matrix

Truth \ Guess	**Human Expert**	**Algorithm**	**Random**
Human Expert	121	124	112
Algorithm	122	126	123
Random	125	121	107

Truth

Guess

The Survey

The DJ

Playlist guess	Guess	Actual
Human Expert	3.6	3.5
Algorithm	3.0	2.25
Random	2.0	3.25

Objective Evaluation

Some playlist stats

Playlist stats

Source	Radio Paradise	Musicmobs	art of the mix	Pandora
Playlists	45,283	1,736	29,164	94
Unique Artists	1,971	19,113	48,169	556
Unique Tracks	6,325	93,931	218,261	908
Average Length	4.3	100	20	11
% with duplicate artist	0.3%	79%	49%	48%
% with consecutive artists	0.3%	60%	20%	5%

Pandora playlist stats based on listening on 44 separate ‘stations’

Objective evaluation
Tag diversity

Playlist Tag Diversity

Source	Tag Diversity	Random
MusicMobs	0.29 / 0.18	0.51 / 0.13
Pandora	0.44 / 0.20	0.64 / 0.19
Art of the mix	0.48 / 0.17	0.61 / 0.11
Radio Paradise	0.75 / 0.13	0.75 / 0.13

Tag Diversity: unique artist tags vs. total artist tags

Radio Paradise diversity examples

Low Diversity Playlists		
Artist	**Track**	**Tags**
Sun Volt	Live Free	**Alt-country**, **americana**, rock, country, **folk**, **indie**
Sun Kil Moon	Gentle Moon	indie, **folk**, singer-songwriter, **americana**, **Alt-country**, alternative
ANi DiFranco	Angry Any More	**folk**, singer-songwriter, female vocalists, **indie**, alternative, rock
Jim White	Handcuffed to a fence in Mississippi	**Alt-country**, singer-songwriter, **americana**, **folk**, **indie**, country
Jess Klein	Soda Water	**folk**, female vocalists, singer-songwriter, **indie**, acoustic, girls with guitars

Diversity: 0.367

11 unique tags out of 30

Radio Paradise diversity examples

High Diversity Playlists		
Artist	**Track**	**Tags**
Big Head Todd & The Monsters	It's Alright	rock, alternative, jam band, prog rock, Jam, 90s
Joni Mitchell	Be Cool	folk, singer-songwriter, female vocalists, Canadian, classic rock, acoustic
Chet Baker	Tangerine	jazz, trumpet, cool jazz, blues, jazz vocals, easy listening

Diversity: 1.0

18 unique tags out of 18

Pandora diversity examples

Low Diversity Playlists		
Artist	**Track**	**Tags**
Project Pitchfork	Timekiller	**industrial, ebm, electronic, darkwave, Gothic, synthpop,**
Covenant	We stand alone	melodic black metal, black metal, synthpop, metal, industrial, futurepop
Icon of Coil	Faith? Not Important	**ebm, industrial, futurepop, electronic, synthpop, darkwave**
Neuroticfish	Waving Hands	**ebm, futurepop, industrial, synthpop, electronic, goth**
Project Pitchfork	Momentum	**industrial, ebm, electronic, darkwave, Gothic, synthpop**
Covenant	Stalker	melodic black metal, black metal, synthpop, metal, industrial, futurepop

Diversity: 0.305

11 unique tags out of 36

Project Pitchfork Radio

Pandora diversity examples

High Diversity Playlists		
Artist	**Track**	**Tags**
Metallica	The Call of Ktulu	metal, thrash metal, heavy metal, rock, hard rock,, metallica
Linkin Park	Pushing Me Away	rock, Nu Metal, alternative, metal, Linkin Park, punk
Creed	One Last Breath	rock, alternative, hard rock, Grunge, metal, punk

Diversity: 0.611

11 unique tags out of 18

Evanescence Radio

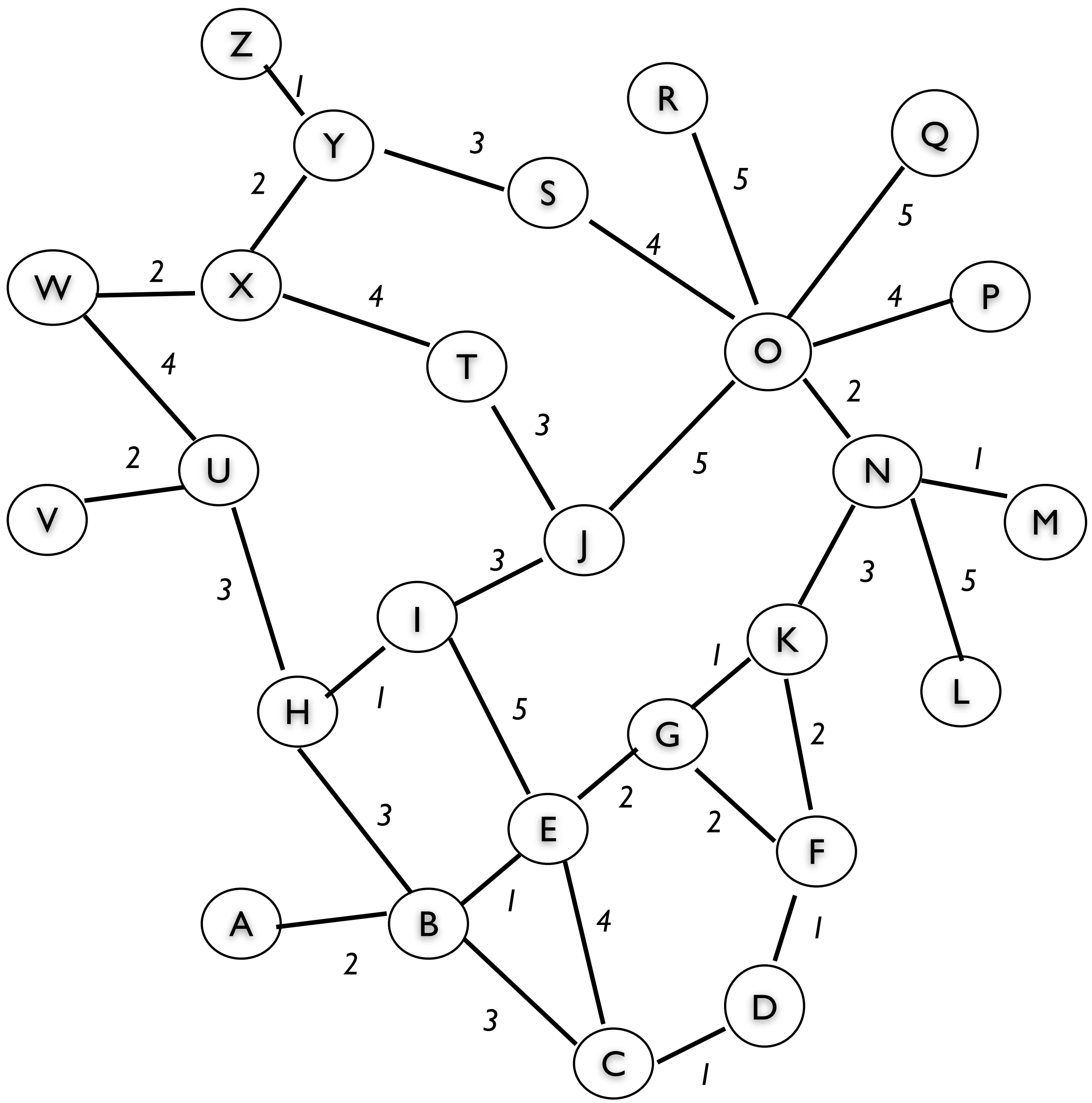
Musicmobs diversity examples

Low Diversity Playlists		
Artist	Track	Tags
Perfect Circle	(54 Tracks)	**rock, alternative, Progressive rock, metal,** hard rock, industrial
Tool	(43 Tracks)	Progressive metal, **Progressive rock, metal, rock, alternative,** Progressive

Diversity: 0.014

8 unique tags out of 582

Playlist Cohesion Metric

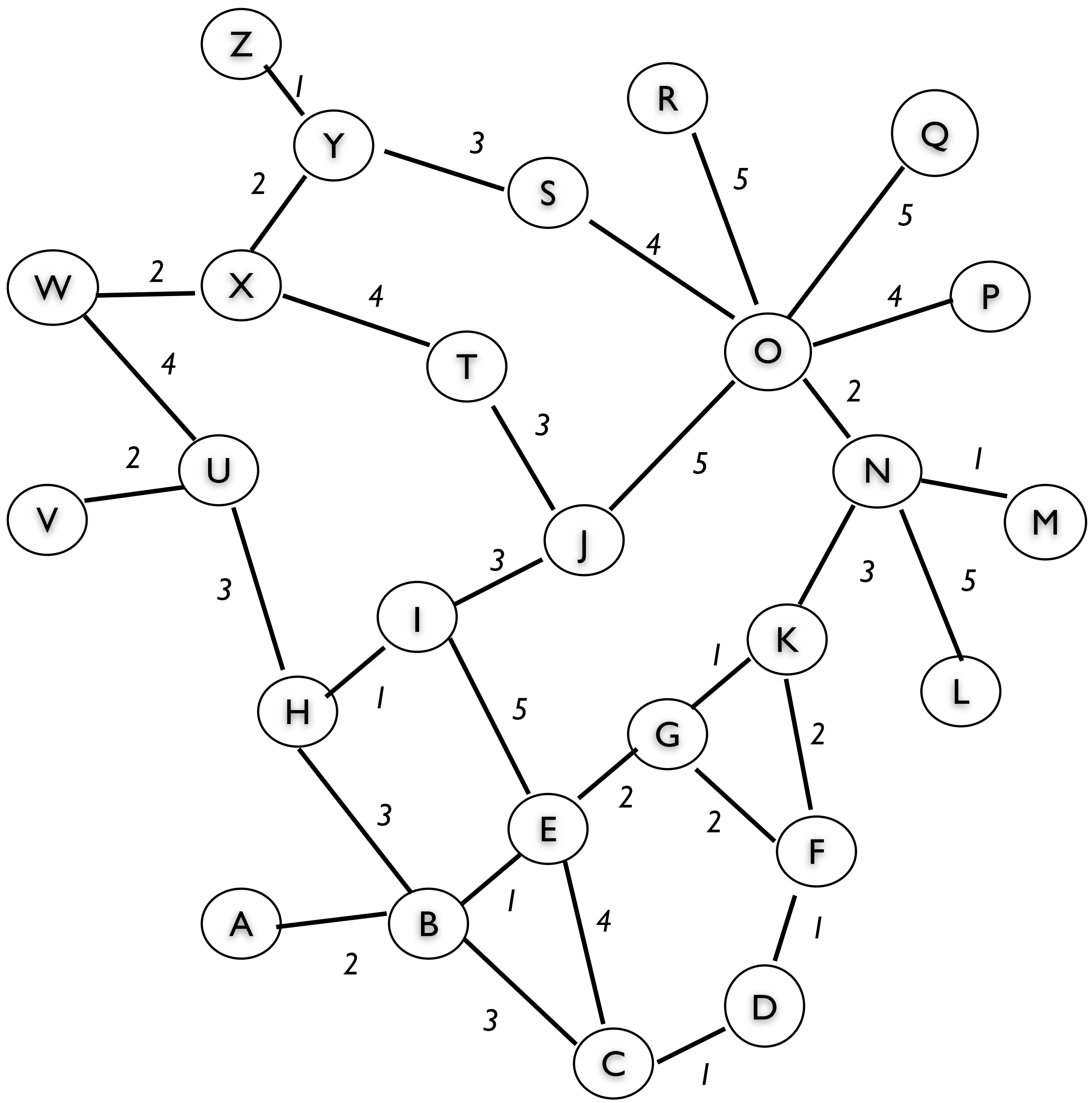


- Goal - find level of cohesion in an ordered sequence such as a playlist
- How:
  - Represent the item space as a connected graph
  - Find the shortest weighted path that connects the ordered sequence
  - Average step length is the cohesion index

Playlist Cohesion Metric

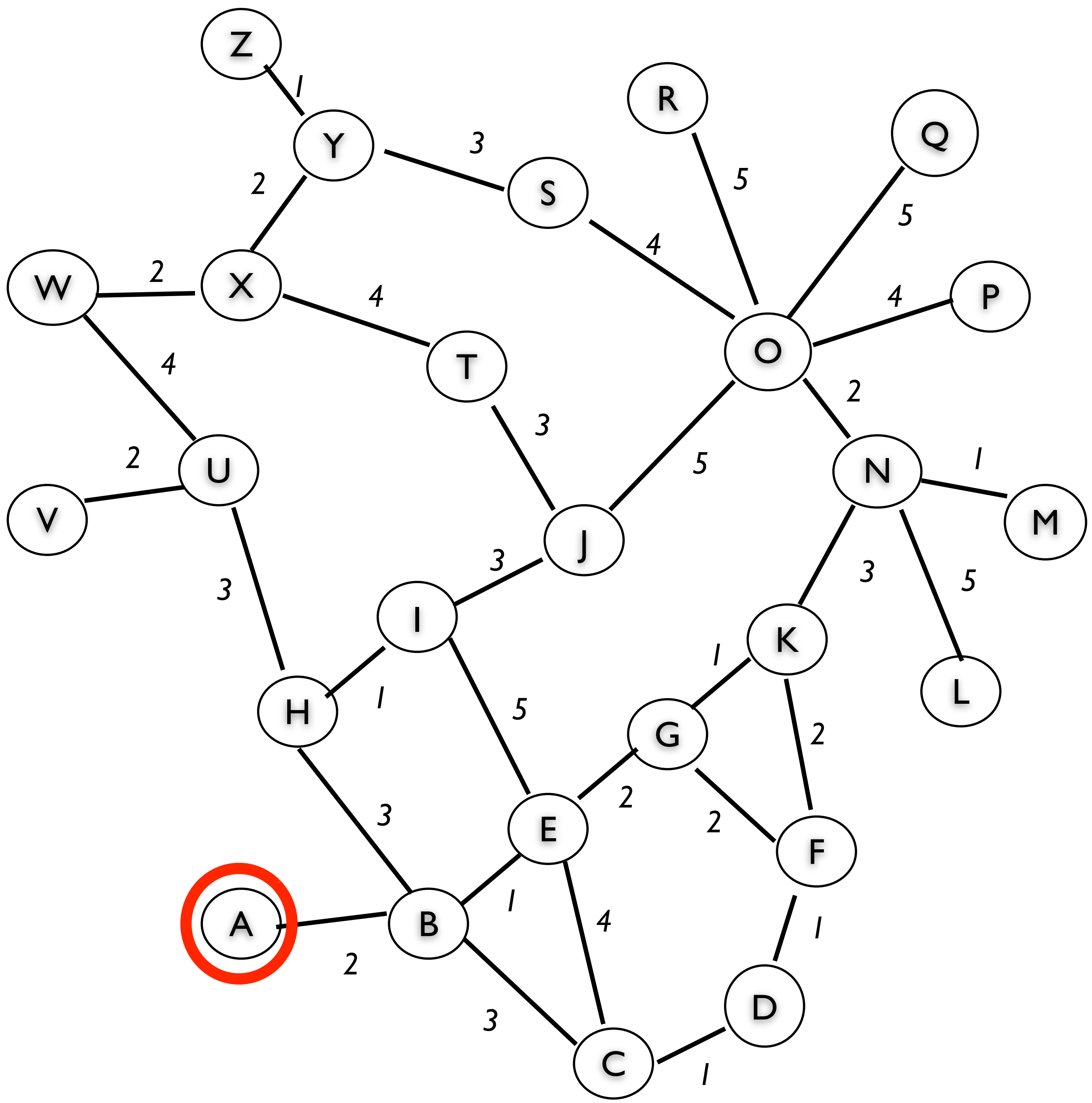
I especially like the "playlist cohesion" metric on slide 185(!) -- i will definitely refer to this next time i make a mix tape for a girl

Playlist Cohesion Metric



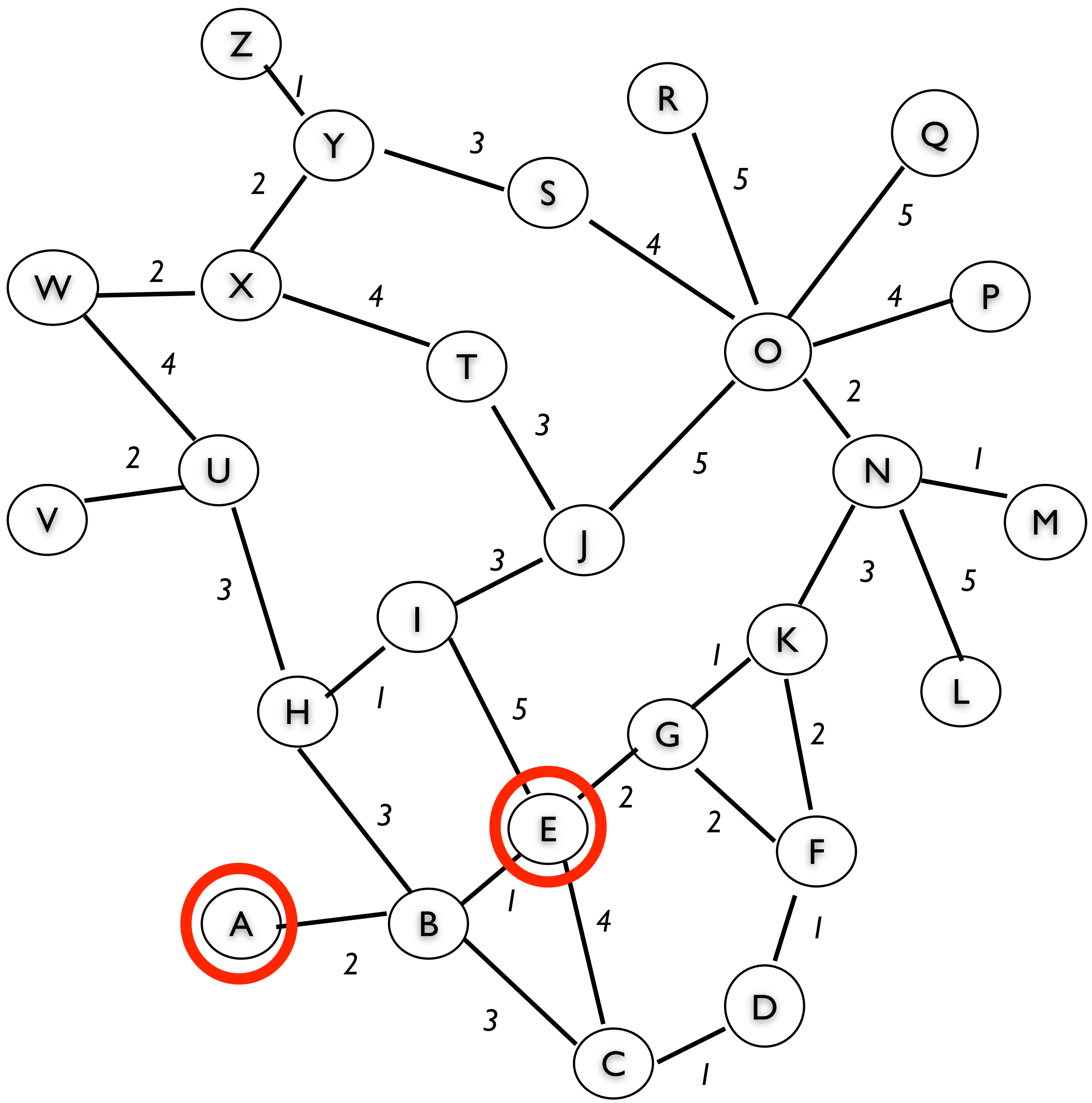
- Consider [A, E, U, X]
- Distance: [3,7,6] = 16
- Average Distance: 5.33

Playlist Cohesion Metric



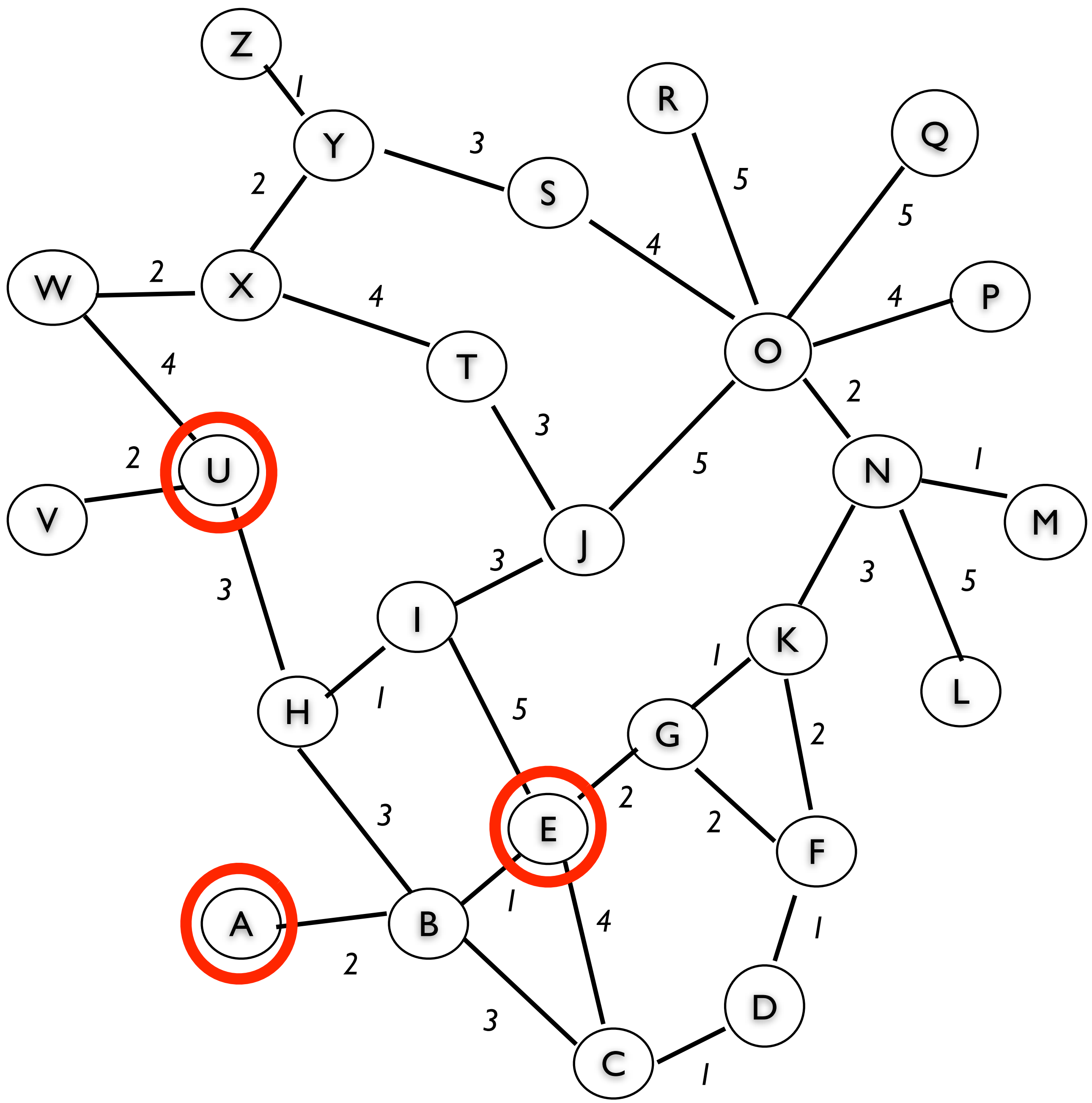
- Consider [A, E, U, X]
- Distance: [3,7,6] = 16
- Average Distance: 5.33

Playlist Cohesion Metric



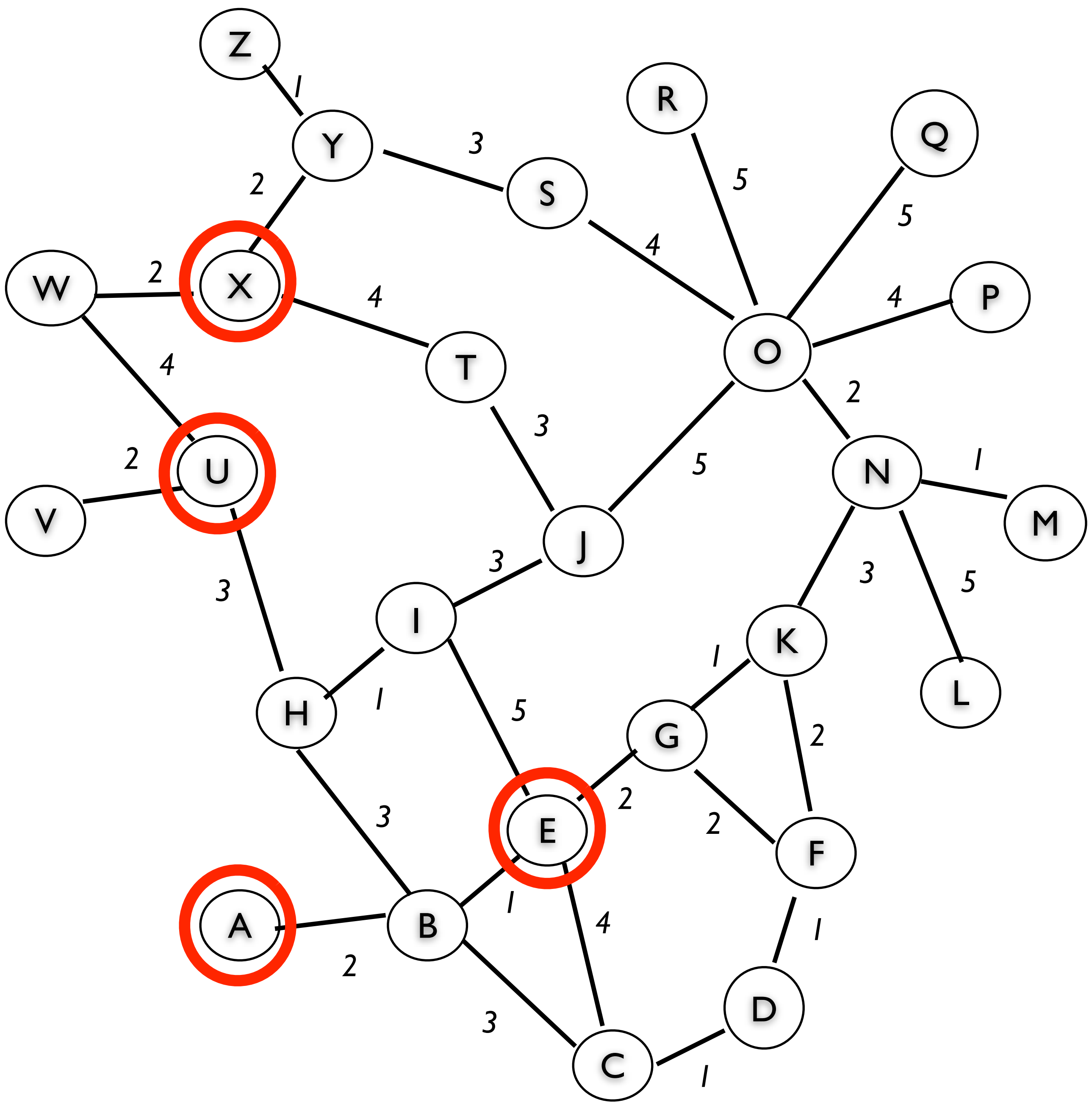
- Consider [A, E, U, X]
- Distance: [3,7,6] = 16
- Average Distance: 5.33

Playlist Cohesion Metric



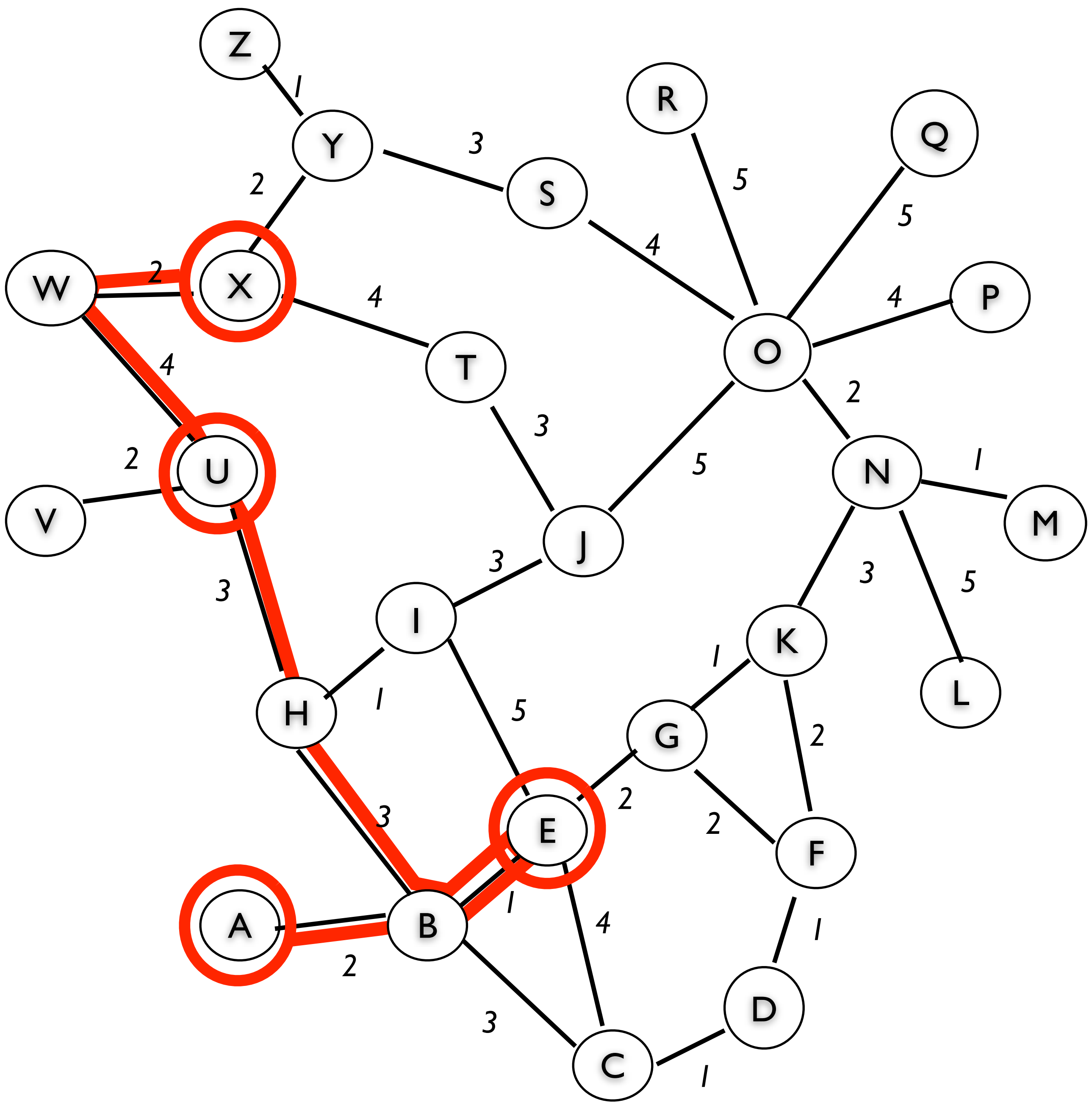
- Consider [A, E, U, X]
- Distance: [3,7,6] = 16
- Average Distance: 5.33

Playlist Cohesion Metric



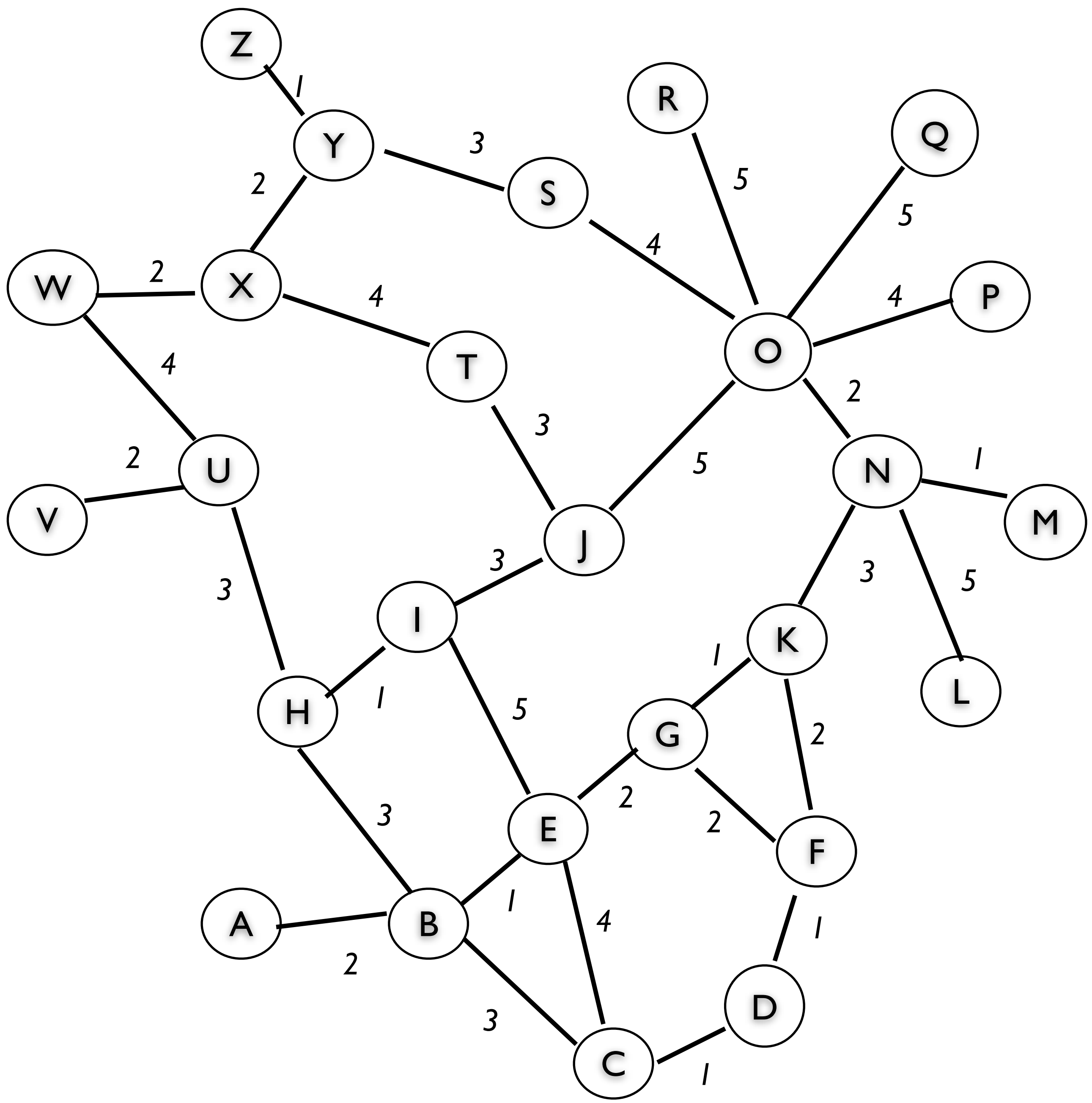
- Consider [A, E, U, X]
- Distance: [3,7,6] = 16
- Average Distance: 5.33

Playlist Cohesion Metric



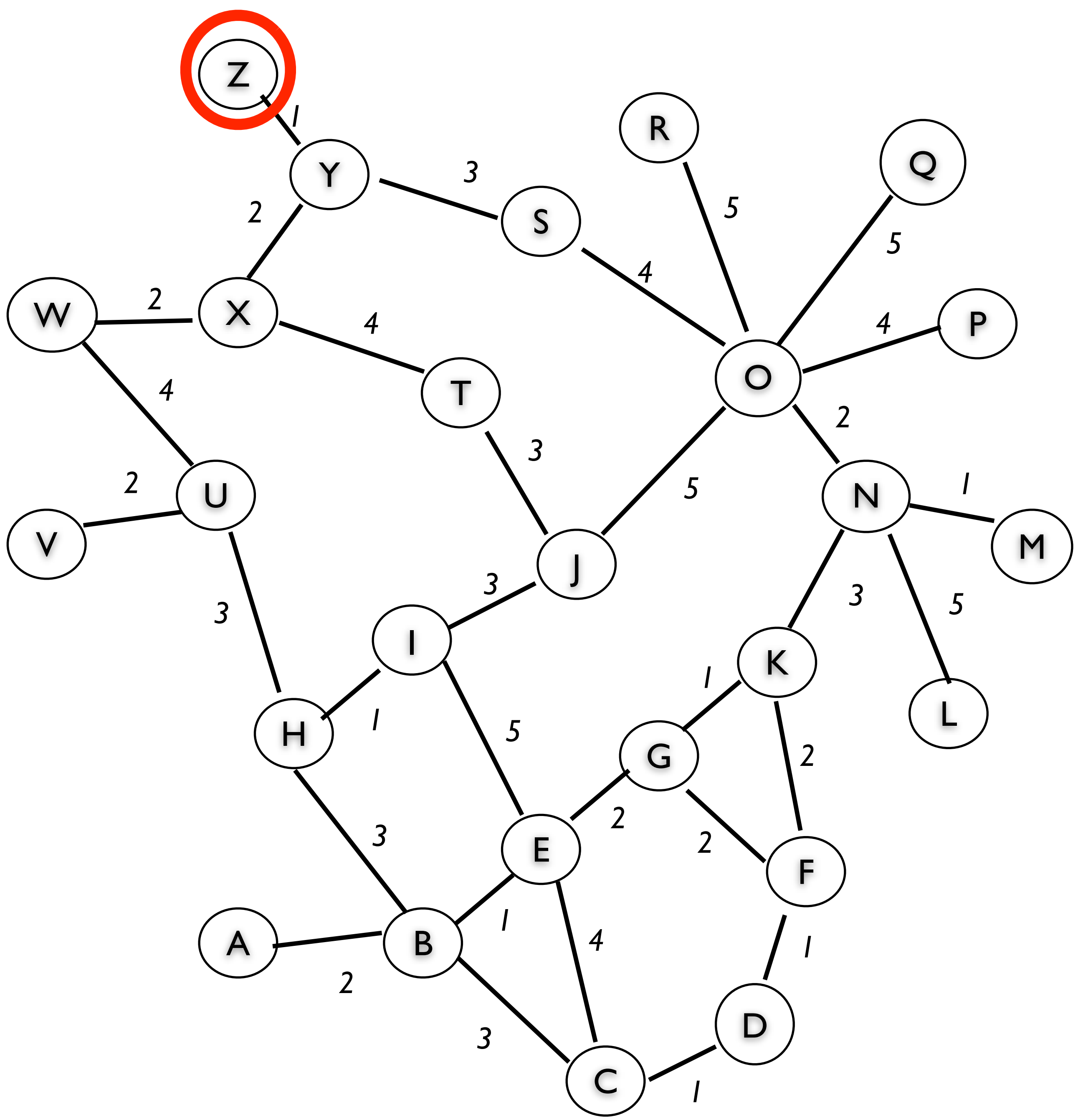
- Consider [A, E, U, X]
- Distance: [3,7,6] = 16
- Average Distance: 5.33

Playlist Cohesion Metric



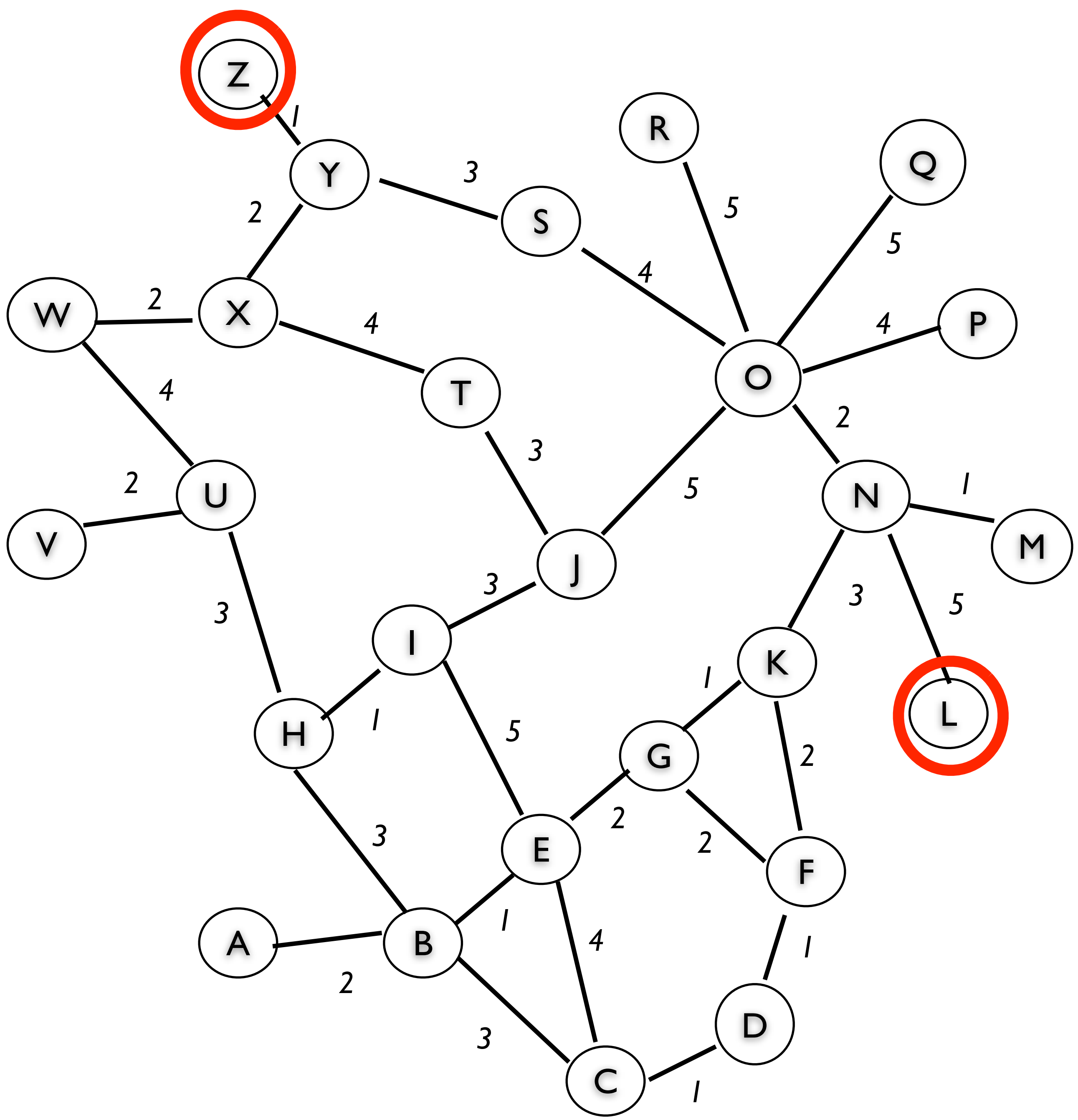
- Consider [Z,L, H, X]
- Distance: [15 , 10 , 9] = 34
- Average Distance: 11.3

Playlist Cohesion Metric



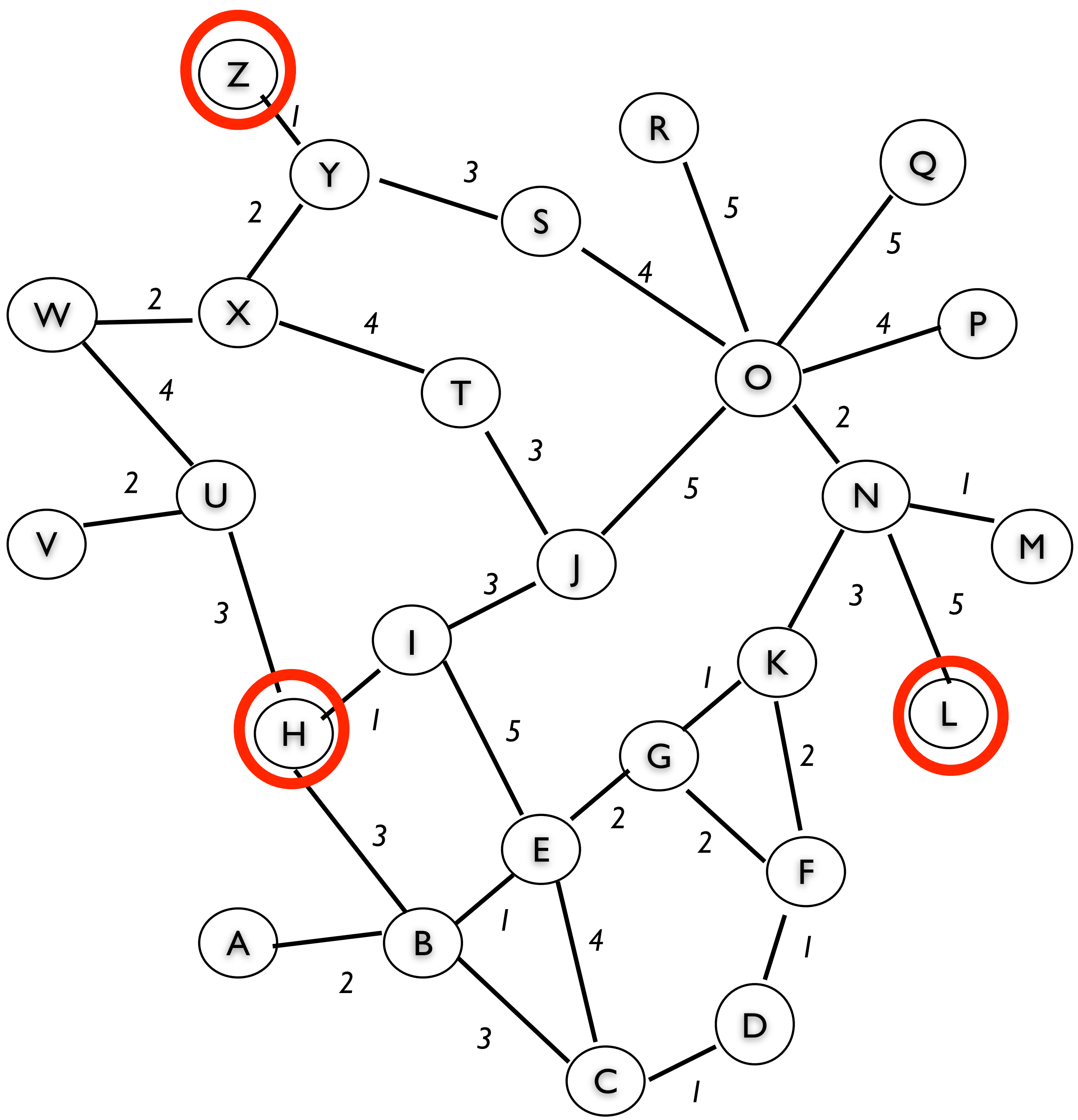
- Consider [Z,L, H, X]
- Distance: [15 , 10 , 9] = 34
- Average Distance: 11.3

Playlist Cohesion Metric



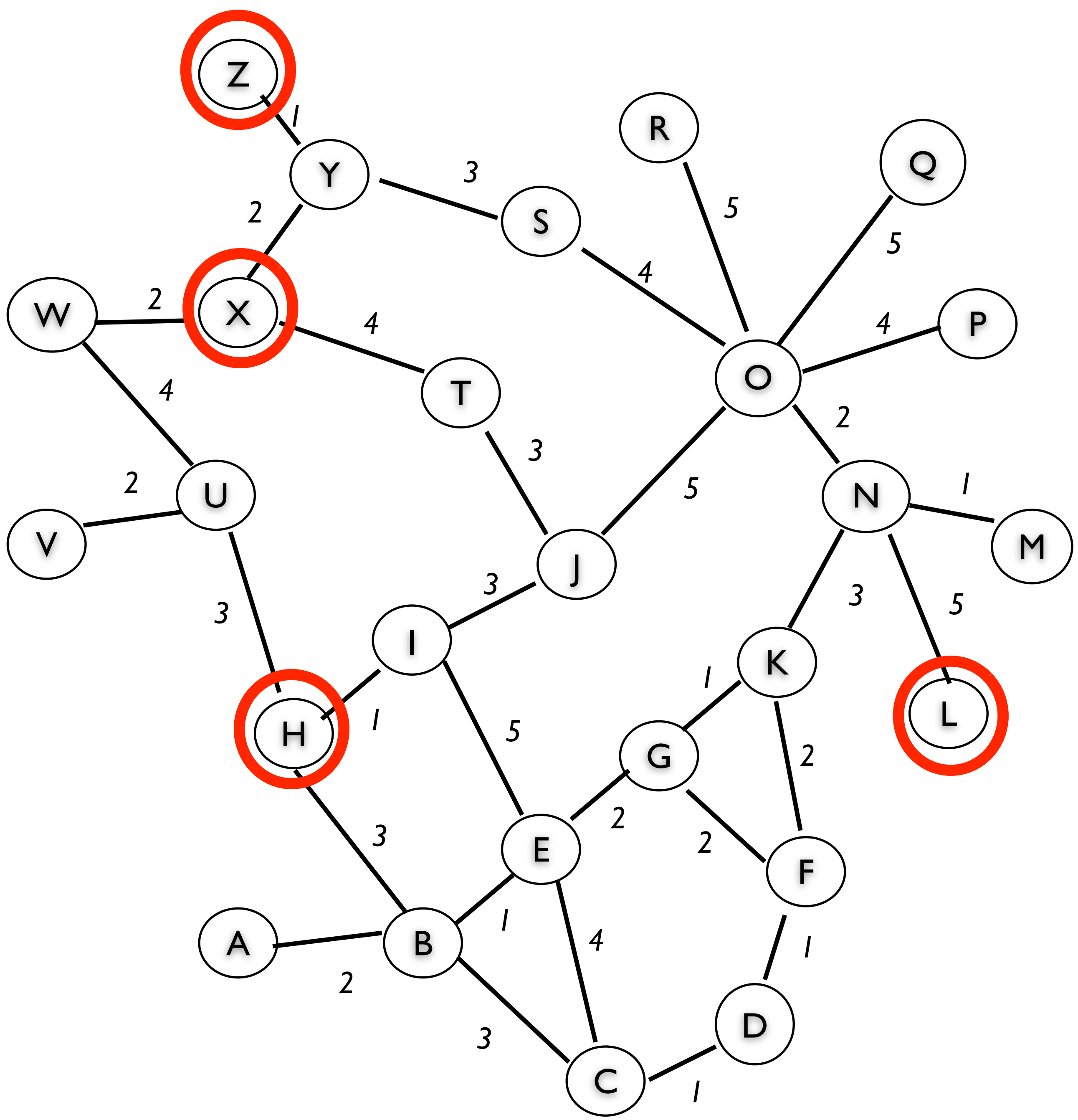
- Consider [Z,L, H, X]
- Distance: [15 , 10 , 9] = 34
- Average Distance: 11.3

Playlist Cohesion Metric



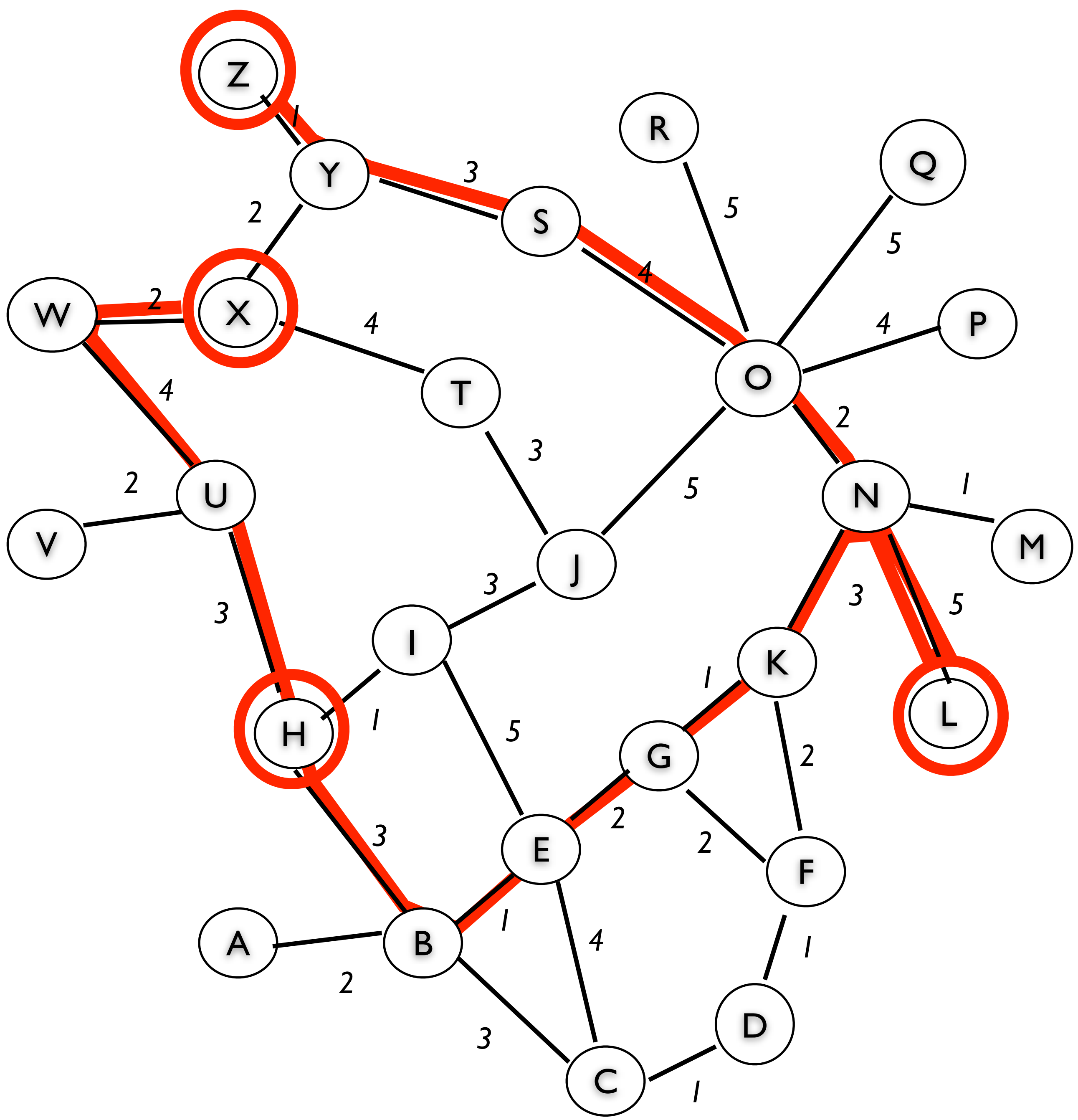
- Consider [Z,L, H, X]
- Distance: [15 , 10 , 9] = 34
- Average Distance: 11.3

Playlist Cohesion Metric



- Consider [Z,L, H, X]
- Distance: [15 , 10 , 9] = 34
- Average Distance: 11.3

Playlist Cohesion Metric



- Consider [Z,L, H, X]
- Distance: [15 , 10 , 9] = 34
- Average Distance: 11.3

Building the graph
MusicBrainz Artist Relations

- Nodes are artists
- Edges are relations, weighted by significance
- 132 Relationship types. some examples:

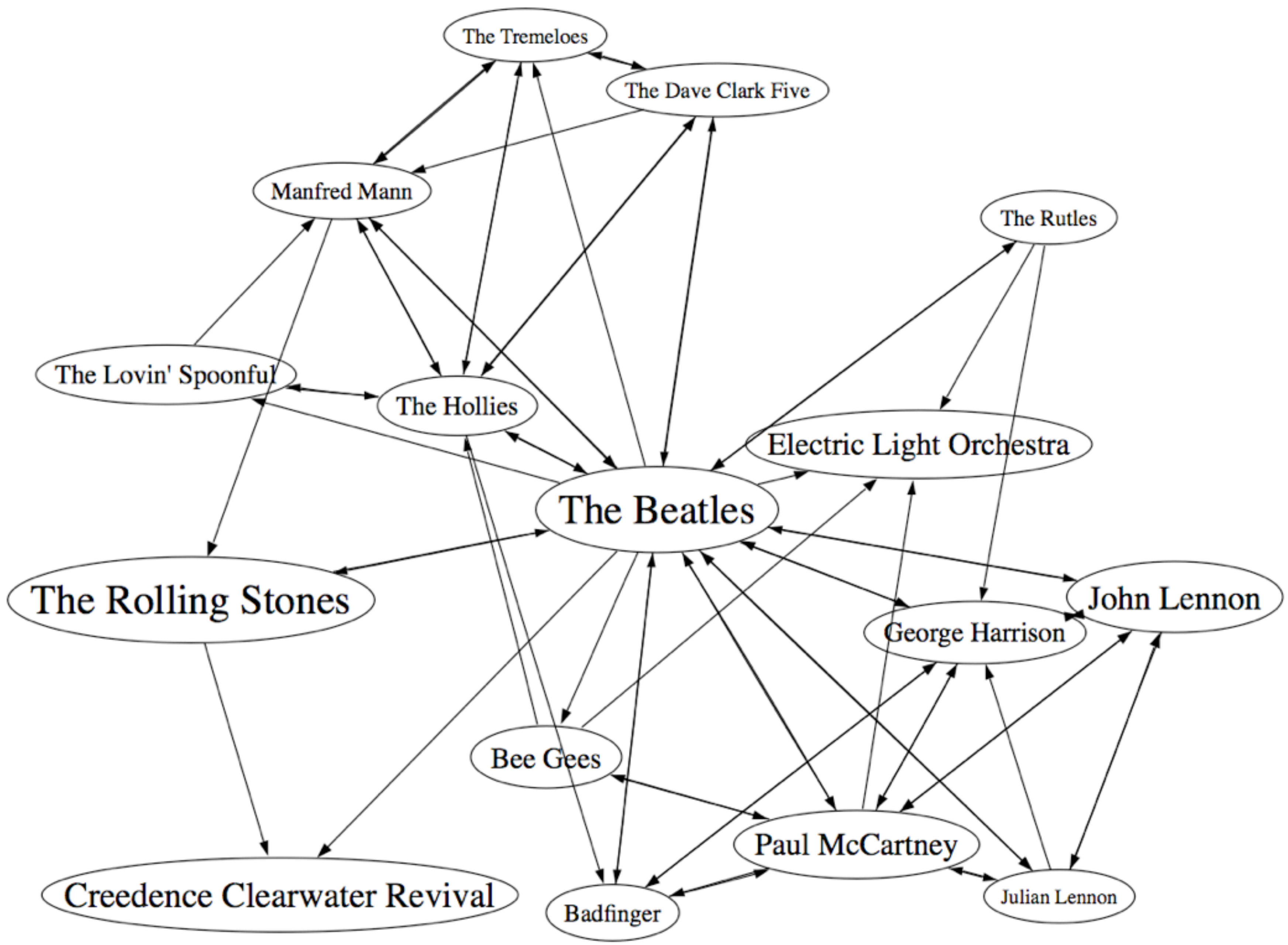
Edge type	Weight
Is Person	1
Member of band	10
Married	20
Performed with	100
Composed	250
Remixed	500
Edited Liner Notes	1000

MusicBrainz Artist Relations Graph

Source	Average inter-song Distance
Radio Paradise	0.08 / 0.06
Pandora	0.11 / 0.12
MusicMobs	0.13 / 0.10
Art of the mix	0.14 / 0.10
Random (RP)	0.27 / 0.22
Random (graph)	0.39 / 0.45
Random (AotM)	0.56 / 0.19

Building the graph
Echo Nest Artist Similarity

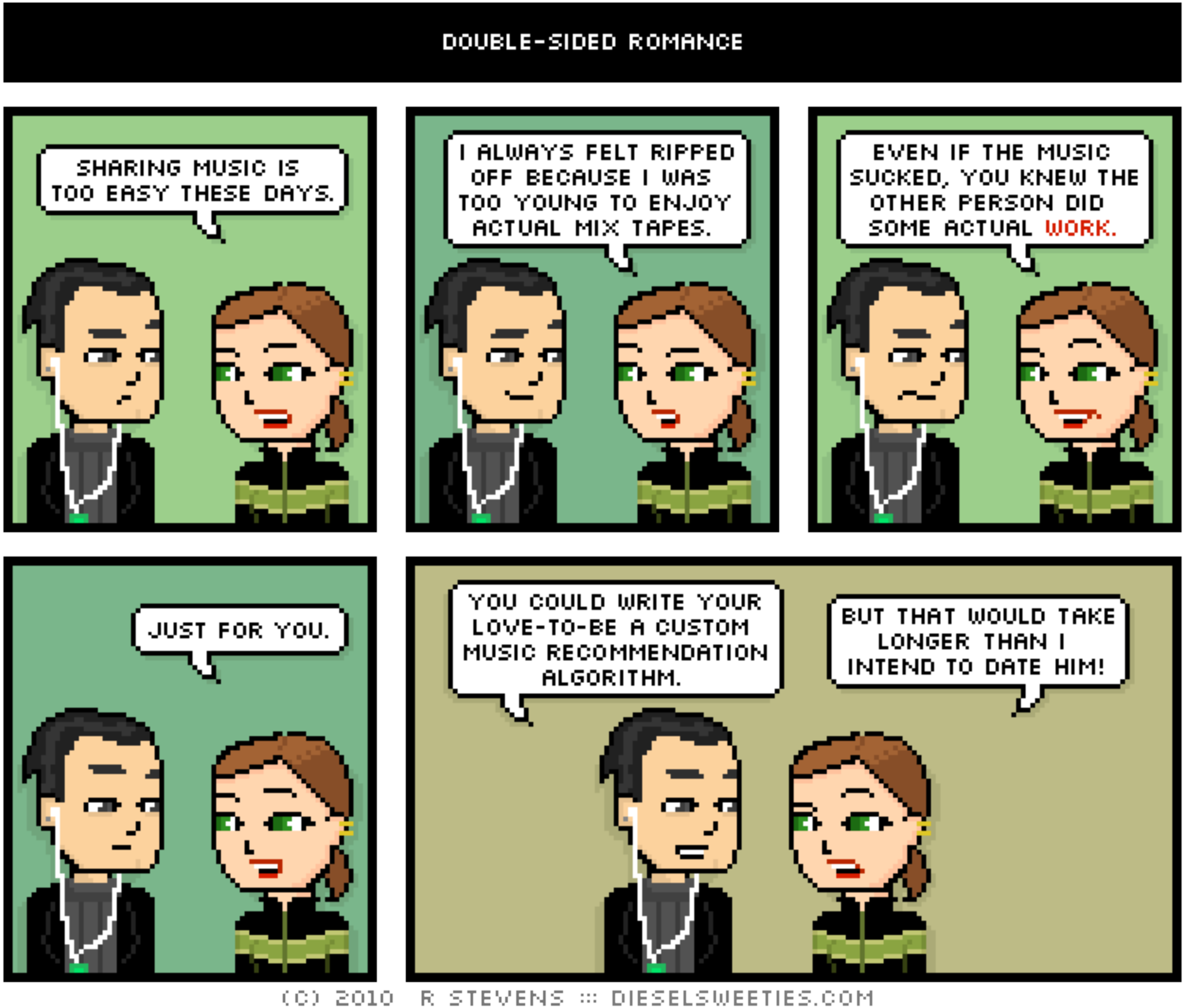
- Nodes are artists
- Edges are similar artists, weighted by similarity



Echo Nest Artist Similarity Graph

Source	Average inter-song Distance
Pandora	1.57 / 1.4
Radio Paradise	2.27 / 1.0
MusicMobs	2.71 / 1.7
Art of the mix	3.02 / 1.4
Random (RP)	4.02 / 1.2
Random (AotM)	7.00 / 1.1
Random (graph)	7.89 / 1.78

The future of playlisting


DOUBLE-SIDED ROMANCE
SHARING MUSIC IS TOO EASY THESE DAYS.
I ALWAYS FELT RIPPED OFF BECAUSE I WAS TOO YOUNG TO ENJOY ACTUAL MIX TAPES.
EVEN IF THE MUSIC SUCKED, YOU KNEW THE OTHER PERSON DID SOME ACTUAL WORK.
JUST FOR YOU.
YOU COULD WRITE YOUR LOVE-TO-BE A CUSTOM MUSIC RECOMMENDATION ALGORITHM.
BUT THAT WOULD TAKE LONGER THAN I INTEND TO DATE HIM!
(C) 2010 R STEVENS :: DIESELSWEETIES.COM

Hybrid Radio

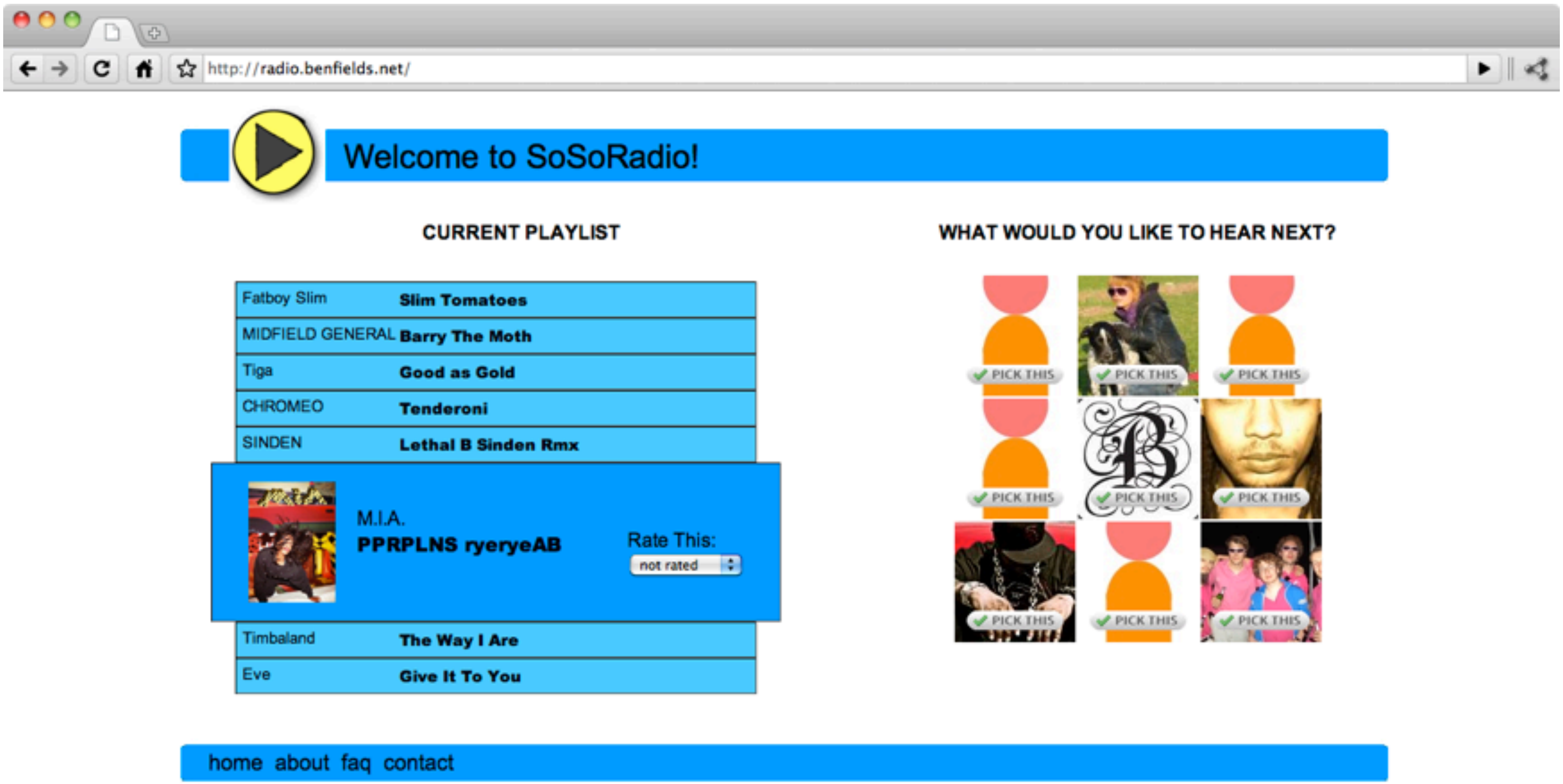
The Social Radio

- Produce playlists via weighted distance paths
- Next destination song is determined via a vote across all listeners
- Candidate songs selected from disparate communities

Hybrid Radio
Ratings

- Ratings are applied to the edge that lead to the song
- Song ratings -> playlist ratings
- Serving 2 purposes
  - Direct evaluation of playlists
  - Object based filtering

Hybrid Radio



Convergence

When the cloud provide all the music and ubiquitous internet provides it all the time recommendation and playlisting merge

Convergence

The celestial jukebox needs a DJ.

THE NEW YORKER

The anonymous programmers who write the algorithms that control the series of songs in these streaming services may end up having a huge effect on the way that people think of musical narrative—what follows what, and who sounds best with whom. Sometimes we will be the d.j.s, and sometimes the machines will be, and we may be surprised by which we prefer

You, the D.J.

Online music moves to the cloud.

by Sasha Frere-Jones

The New Yorker, June 14, 2010