

# Cooperative Music Management and Scheduling

Jan Krämer

Media Computing Group  
Dept. of Computer Science  
RWTH Aachen University

Presentation at Club i10 on 27.7.2006

# A Short Introduction

## ***Name:***

Jan Krämer

## ***Field of Study:***

Computer Science, Psychology

## ***Hobbies:***

College Radio

# My Project in 30 Seconds

What I am doing is...

Music Management

# My Project in 30 Seconds

**What I am doing is...**

Music Management and Scheduling

# My Project in 30 Seconds

**What I am doing is...**

Cooperative Music Management and Scheduling

# My Project in 30 Seconds

## What I am doing is...

Cooperative Music Management and Scheduling

## I am doing this because...

...experienced first hand how much problems people had.

# Outline

1 Background

2 Research

3 My Work

# Outline

## 1 Background

## 2 Research

## 3 My Work



# Structure of a Radio Station

## Groups

- Management and support staff
- On-air editorial staff
- Music editorial staff

# Structure of a Radio Station

## Groups

- Management and support staff
- On-air editorial staff
- Music editorial staff

# Music Selection for Radio

## Some Concepts

- Songs
- Rotation groups
- Clocks



# Music Selection for Radio

## Some Concepts

- Songs
- Rotation groups
- Clocks
- Stop sets
- Drops, jingles, teasers...

# Challenges

## To Schedule Music You Need:

- Songs you can schedule  
**But:** Keeping track of the collection is hard
- Information about those songs  
**But:** What information do you need? Who enters the data?
- Specification of the playlist  
**But:** How and in what detail do you want to specify it?

# Challenges

## To Schedule Music You Need:

- Songs you can schedule  
**But:** Keeping track of the collection is hard
- Information about those songs  
**But:** What information do you need? Who enters the data?
- Specification of the playlist  
**But:** How and in what detail do you want to specify it?

# Challenges

## To Schedule Music You Need:

- Songs you can schedule  
**But:** Keeping track of the collection is hard
- Information about those songs  
**But:** What information do you need? Who enters the data?
- Specification of the playlist  
**But:** How and in what detail do you want to specify it?



# Outline

1 Background

**2 Research**

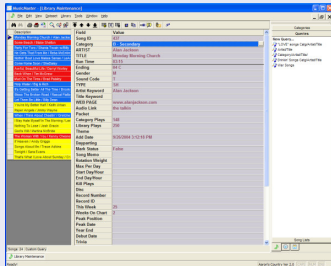
3 My Work



# Commercial

## Software for Radio Stations

- Either very simple (for example Raduga)...
- ...or very complex (for example MusicMaster)...



# Commercial

## Software for Radio Stations

- Either very simple (for example Raduga)...
- ...or very complex (for example MusicMaster)...
- ...but there is no transition path.

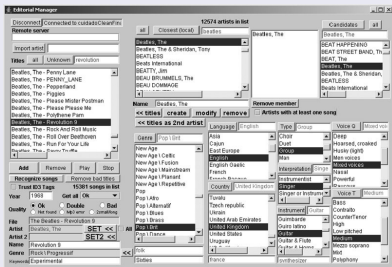
?

## Related Work

## Scientific

## “Sony Music Browser” by Francois Pachet et al. 2004

- Constraint-based playlist generation
- Shared metadata (via server2server communication)
- Central server



# Scientific

## Playlists Based on Skipping by Pampalk et al. 2005

- Music Similarity based on Spectrum, timbre and rhythm
- Playlistgeneration based on seed song and skipping behaviour

# Scientific

## Lots of Work on Feature Extraction

- Example: Audio Summarization by Peeters et al. 2002
- Example: Perceptual Tempo by McKinney and Moelants 2004

# Method for the Preparatory Study

## Based on Contextual Inquiry

- On-Site
- Master/Apprentice



# Target Users

## Target Users

Non-professional music editorial staff

# Target Users

## Target Users

Non-professional music editorial staff

## Participants

- Five music editors from four college radio stations
- A telephone interview with head of music programming Radio NRW

# Results From a Preparatory Study

## Findings Include:

- Music editors spend most time managing the collection and checking playlists
- Editors work around their software a lot
- Being able to listen to music and playlists is extremely important
- Editing playlists is painful

# Outline

1 Background

2 Research

**3 My Work**

# The Basic-Concepts Slide

## *Concepts*

- Identity (persons, groups and services)
- Objects, i.e. songs, artists, recordings
- Tags, ratings, fields, relationships, note
- Revision control
- Constraints
- License of information

# Use Case: Updating the Collection

## The Collection Interface

✕

### Collection of Hochschulradio Aachen

	Song	Artist	Album	Speed	Rock
<b>Add Music</b>	Lorem Ipsum	dolor silt	amet consectetur		
	Lorem Ipsum	dolor silt	amet consectetur		
<b>Filter</b>	Only Happy	Garbage	Garbage	2	
	Lorem Ipsum	dolor silt	amet consectetur		
	Lorem Ipsum	dolor silt	amet consectetur		
	Lorem Ipsum	dolor silt	amet consectetur	5	
	Lorem Ipsum	dolor silt	amet consectetur		
	Lorem Ipsum	dolor silt	amet consectetur		
	Lorem Ipsum	dolor silt	amet consectetur	2	
	Lorem Ipsum	dolor silt	amet consectetur	4	
	Lorem Ipsum	dolor silt	amet consectetur	1	
	Lorem Ipsum	dolor silt	amet consectetur		
Lorem Ipsum	dolor silt	amet consectetur	3		

**Only Happy When It Rains**  
of: **Garbage**

alternative  
alternative rock      aggressiveness: 2/5  
female voice            speed:3/5  
grunge  
rock

# A Quick Tour

## Use Case: Verifying the Playlist

**Playlist Thursday 27.7.2006**

11:35	Lorem Ipsum - Dolor Sit Amet (2:45)
11:40	Consectetur - Adipiscing Elit (2:55)
11:45	Vestibulum Porttitor - Ipsum (3:33)
11:50	Eu Nulla - Nam Dapibus (3:15)
11:50	<b>Only Happy When It Rains - Garbage (3:46)</b>
11:55	Risus - At Quam (4:05)
12:00	Song of the Week at 11:50
12:00	Morbi A - Est Praesent (1:48)
12:05	Sed - Mi Quis Ipsum (3:10)
12:05	Rutrum Pretium - Fusce (2:16)
12:10	Song of the Week at 12:00

**Only Happy When It Rains**  
of: **Garbage**

alternative rock      aggressiveness: 2/5  
alternative rock      female voice      speed: 3/5  
grunge  
rock

Talk Time      Chorus

# A Quick Tour

## Use Case: Updating the playlist structure

**Rules for Every Monday**

Timeline (11:35 to 12:10):

- 11:35 - 11:55: No Artist Rule (Purple)
- 11:55 - 12:00: No Boyband Rule (Blue)
- 12:00 - 12:05: Song Of The Week Rule (Green)
- 12:05 - 12:10: (Light Blue)

**Song Of The Week**

- Is Exclusive
- Is Unbreakable

Logic Tree:

- Has Tag[Song Of the Week]
- AND
- Has Tag[Playable]
- AND
- OR



# Used Technologies

- C++
- Qt 4.2
- Peer2Peer communication, DHT over HTTP
- WebServices

# The Method of Evaluation

## Deployment

- The system will be deployed at the college radio in Aachen
- Documentation as case-study

Any Questions?

Any Questions?

**P.S.**

I need a name....