

# Эффективность и оптимизация извлечения характеристик из символьной музыки

Чесноков Александр Михайлович

Научный руководидитель: Ковалев Дмитрий Андреевич, руководитель направления экспериментальных систем машинного обучения, SberDevices

Московский физико-технический институт

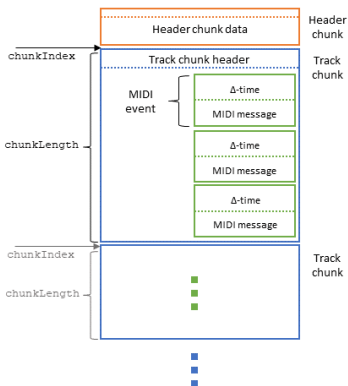
17 мая 2024



# Содержание

- Введение
- Работа за семестр
- Результаты и дальнейшие планы

# Символьная музыка



Структура формата  
MIDI.

# jSymbolic by Cory McKay

- Pitch Statistics
- Melodic Intervals
- Rhythm
- Instrumentation
- Texture
- Dynamics
- Всего 246 фичей или 1497 значений с учетом гистограмм

# Проблемы и задачи

Проблемы:

- Плохие решения в архитектуре кода

# Проблемы и задачи

## Проблемы:

- Плохие решения в архитектуре кода
- Критические баги

# Проблемы и задачи

## Проблемы:

- Плохие решения в архитектуре кода
- Критические баги
- Неэффективное вычисление фичей

# Проблемы и задачи

## Проблемы:

- Плохие решения в архитектуре кода
- Критические баги
- Неэффективное вычисление фичей

## Задачи:

- Исправить архитектуру кода для более упрощения работы



# Проблемы и задачи

## Проблемы:

- Плохие решения в архитектуре кода
- Критические баги
- Неэффективное вычисление фичей

## Задачи:

- Исправить архитектуру кода для более упрощения работы
- Изучить производительность вычислений

# Проблемы и задачи

## Проблемы:

- Плохие решения в архитектуре кода
- Критические баги
- Неэффективное вычисление фичей

## Задачи:

- Исправить архитектуру кода для более упрощения работы
- Изучить производительность вычислений
- Пересмотреть подход к вычислению фичей и оптимизировать их вычисление

# Диаграммы кода

- Рефакторинг кода для упрощения работы с кодом
- Созданы диаграммы кода для лучшего понимания. (См. приложение [1,2,3])

# Оптимизации

- Работа с уменьшением разрешения музыки. Подробнее в приложении [4]

# Оптимизации

- Работа с уменьшением разрешения музыки. Подробнее в приложении [4]
- Изменение логики работы

# Оптимизации

- Работа с уменьшением разрешения музыки. Подробнее в приложении [4]
- Изменение логики работы
- Мелкие технические оптимизации кода

# Бенчмарк. Что именно тестируем

- Датасет из 1000 файлов

# Бенчмарк. Что именно тестируем

- Датасет из 1000 файлов
- Режимы тестов
  - Обработка всех фичей



# Бенчмарк. Что именно тестируем

- Датасет из 1000 файлов
- Режимы тестов
  - Обработка всех фичей
  - Обработка всех фичей, кроме гистограмм

# Бенчмарк. Что именно тестируем

- Датасет из 1000 файлов
- Режимы тестов
  - Обработка всех фичей
  - Обработка всех фичей, кроме гистограмм
  - Обработка каждой фичи отдельно

# Бенчмарк. Техническая часть

- Java Microbenchmark Harness (JMH) [Ссылка на репозиторий]
- RAM: 16 Гб, vCPU: 4

# Результаты

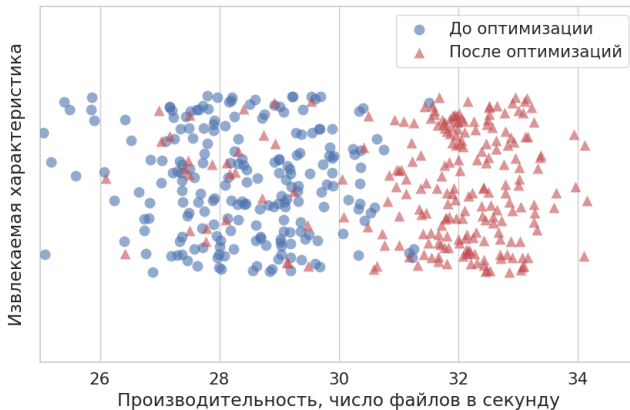


Рис. 1, результаты замеров

# Дальнейшая работа

Бенчмарки:

- Тесты для файлов большей длины

# Дальнейшая работа

Бенчмарки:

- Тесты для файлов большей длины
- Увеличить число обрабатываемых файлов для тестов

# Дальнейшая работа

Бенчмарки:

- Тесты для файлов большей длины
- Увеличить число обрабатываемых файлов для тестов
- Новые параметры, от которых может зависеть время работы

# Дальнейшая работа

## Бенчмарки:

- Тесты для файлов большей длины
- Увеличить число обрабатываемых файлов для тестов
- Новые параметры, от которых может зависеть время работы

## Оптимизация

- Введение многопоточности



# Дальнейшая работа

## Бенчмарки:

- Тесты для файлов большей длины
- Увеличить число обрабатываемых файлов для тестов
- Новые параметры, от которых может зависеть время работы

## Оптимизация

- Введение многопоточности
- Построение автомата вычисления фичей

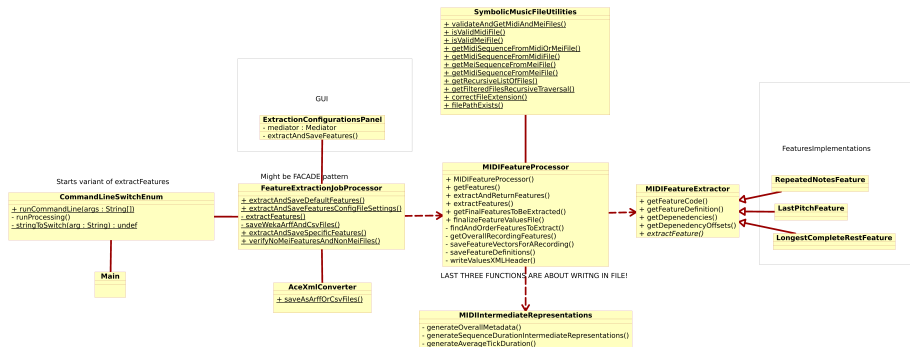
# Литература

- 1 Репозиторий проекта <https://github.com/chesnokoff/jSymbolic>
- 2 Cory McKay, Julie E. Cumming, Ichiro Fujinaga «JSYMBOLIC 2.2: EXTRACTING FEATURES FROM SYMBOLIC MUSIC FOR USE IN MUSICOLOGICAL AND MIR RESEARCH»
- 3 Federico Simonetta, Ana Llorens, Martín Serrano, Eduardo García-Portugués, Álvaro Torrente «OPTIMIZING FEATURE EXTRACTION FOR SYMBOLIC MUSIC» (Сравнительный анализ библиотек)
- 4 [ccrma.stanford.edu](http://ccrma.stanford.edu): Standard MIDI File Structure

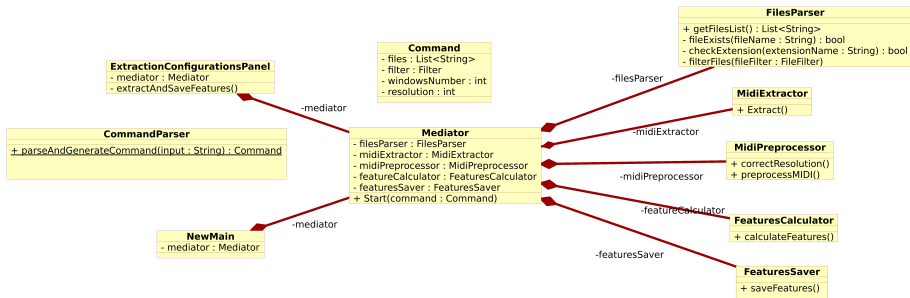
# Литература

- 1 Robert C. Martin, «Clean Code», 2008
- 2 [majicdesigns.github.io](https://majicdesigns.github.io/): MIDI Beat Time Considerations (Продолжительность тиков в MIDI)
- 3 Репозиторий JMH

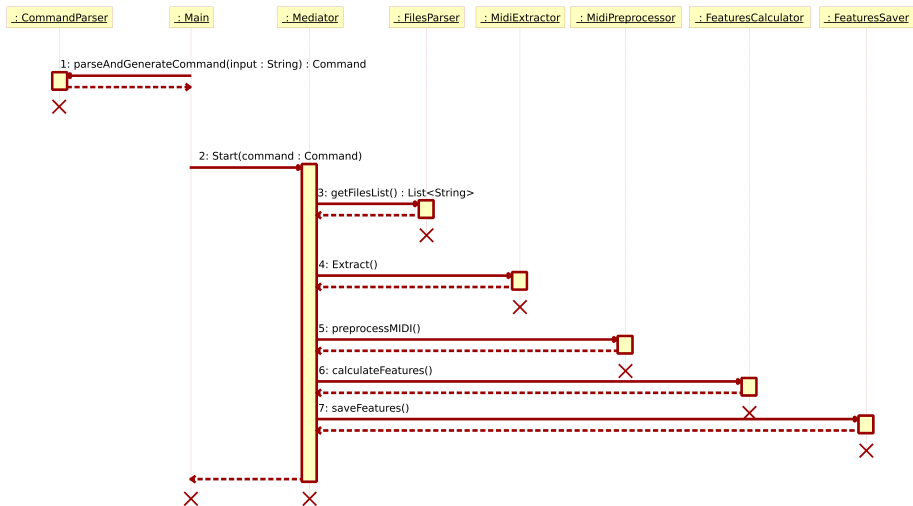
# Приложение 1: Классовая диаграмма до изменений



# Приложение 2: Ожидаемая классовая диаграмма



# Приложение 3: Ожидаемая поведенческая диаграмма



## Приложение 4: Работа с уменьшением разрешения

- **Проблема:** библиотека при обработке музыки выделяет память, пропорциональную количеству тиков в файле. Однако если тиков много, то вылетает Out Of Memory
- **Решение:** уменьшить разрешение музыки. Можно снизить разрешение, так как человеческий слух воспринимает аудиосигнал продолжительностью не менее **250 микросекунд**.

[Разбор биологических статей на данную тему]

## Приложение 5: Некоторые результаты замеров

**Table:** Таблица 1. Производительность до и после оптимизаций

Характеристика	До оптимизаций, файлов/сек.	После оптимизаций, файлов/сек.
Chord-Duration	29.355888	32.837083
First-Pitch	27.583031	32.637643
Average-Note- Duration	27.720014	31.716909
Repeated-Notes	28.320446	31.783939
Number-of-Pitch- Classes	30.326728	32.471622