

Доклад

На тему «Каталогизация в музыке»

Студента второго курса магистратуры заочного отделения

По направлению «информационные технологии в музыке и саунд-дизайн»

Иванова Григория Ильича

Каталогизация в музыке — это упорядочение музыкальных произведений, записей или коллекций путем создания систематизированного списка (каталога) с метаданными. Она включает фиксацию авторов, исполнителей, жанров, уникальных идентификаторов (ISRC, ISWC) и правовых данных. Процесс помогает управлять фонотеками, защищать права и искать записи.

Хотелось бы рассмотреть, с помощью каких средств можно заниматься каталогизацией. Если вы работаете с большим количеством музыкальных треков, занимаетесь каталогизацией или построением библиотеки звуковых эффектов, то в этом случае можно воспользоваться универсальной системой звуковых категорий UCS (Universal Category System).

UCS — это открытый стандарт (open-source) для формирования имен звуковых файлов (изначально - звуковых эффектов), разработанный Тимом Нильсеном, звукорежиссером студии Skywalker Sound. Ключевая особенность стандарта UCS — это максимальная информативность имен файлов, куда можно «зашивать» такие данные как: Категория/Подкатегория (Category/Sub-Category), Дополнительные категории (VendorCategory и UserCategory), Название трека (TrackName), автора трека (CreatorID) и, наконец, Альбома (Библиотеки) (SourceID), в который входит данный трек.

Также в UCS допускается использование Пользовательских Данных (UserData) в конце имени файла, которые могут пригодиться, например, для присвоения уникального идентификатора трека в базе данных.

На данный момент (2023 год) раздел UCS для музыки не является исчерпывающим для описания всех возможных типов и жанров музыкального материала, однако вполне достаточен для их минимально-информативного описания. А сочетание двух Дополнительных категорий (VendorCategory и UserCategory), которые добавляется в имя файла после Категории/Подкатегории UCS, позволяет добиться приемлемого уровня наглядности и информативности о музыкальном содержании треков, их авторах, альбомах и т.п.

Столь наглядные и информативные имена файлов позволяют уверенно ориентироваться в файловых массивах, а также использовать специальное программное обеспечение для каталогизации звуковых треков, которое поддерживает стандарт UCS (Soundminer, Basehead, Soundly и т.п.). Эти программы могут автоматически извлекать метаданные из всех блоков информации стандарта UCS, заносить их в столбцы, согласно правилам, которые задает пользователь программы.

Применение UCS позволяет стандартизировать структуру имен файлов, что не дает файловому массиву превратиться в «помойку». Зачастую подобное происходит, когда большое количество авторов по-разному присваивают имена своим трекам и закладывают в них разную информацию.

Структура имен файлов стандарта UCS

Стандарт UCS рекомендует использовать специальную структуру для имен файлов. При соблюдении этой структуры можно пользоваться скриптами в различных поисковых программах, которые позволяют автоматически анализировать, расшифровывать и переносить информацию из имени файла в различные поля или столбцы метаданных. Кроме того, UCS наглядно отображает информацию в самом имени файла, что повышает читаемость и понимание содержимого данного трека даже не слушая его.

Имя файла по стандарту UCS отвечает на следующие вопросы:

К какой категории и подкатегории относится этот трек?

Что это за трек и какое у него примерное содержимое?

Кто создал или издал этот трек?

Для какого проекта или библиотеки был создан или записан этот трек?

Базовый вариант имени файла по стандарту UCS состоит из четырех блоков информации и каждый из этих блоков должен быть отделен символом нижнего подчеркивания «_».

Нижнее подчеркивание «_» — это зарезервированный символ и может использоваться только для разделения между собой блоков информации стандарта UCS.

Последовательность четырех обязательных блоков в имени файла выглядит следующим образом:

CatID_TrackName_CreatorID_SourceID

CatID - Сокращенное сочетание Категории и Подкатегории UCS (смотрите таблицу с полным списком CatID здесь). Например, CatID получившимся из сочетания Категории MUSIC и Подкатегории SONG будет MUSCSong

TrackName - Краткое описание или название трека (желательно не более 25 символов)

CreatorID - Звукорежиссер, музыкант или издатель трека (для уменьшения длины имени файла можно использовать аббревиатуры для имен авторов, например PPetr -> Petr Petrov или ASer -> Alexander Sergeev)

SourceID - Название проекта, альбома или библиотеки (либо аббревиатура, по аналогии с CreatorID) к которому принадлежит данный трек.

Помимо четырех обязательных блоков информации в стандарте UCS существуют три дополнительных блока, которые необязательны к заполнению, но могут помочь в уточнении основных блоков или для хранения дополнительной информации.

CatID(-UserCategory)_(VendorCategory-)TrackName_CreatorID_SourceID_UserData

UserCategory - Необязательное уточнения блока CatID. Его задача — дальнейшее пояснение Категории/Подкатегории при помощи дополнительной информации. Например, в него можно поместить стиль трека: TRAILER, EPIC, HIPHOP. Отделяется от CatID при помощи дефиса.

VendorCategory - Дополнительный блок информации, который может дополнить блок TrackName. Этот блок следует сразу после CatID (в сочетании с UserCategory) и отделяется нижним подчеркиванием. Для отделения от следующего за ним названия трека (TrackName) после блока VendorCategory используется знак дефиса «-».

UserData - Этот блок информации имеет свободную форму и может быть использован по желанию пользователя, например, для уникального идентификатора базы данных или дальнейшего уточнения информации по треку. Отделяется от блока SourceID при помощи нижнего подчеркивания

Детальное описание блоков имени файла формата UCS

CatID — это сокращённое (аббревиатура) сочетание Категории и Подкатегории, которое позволяет уменьшить длину имени файла. Следует помнить, что CatID — это жестко прописанная структура, которую вы можете посмотреть в списке Категорий/Подкатегорий UCS здесь и она не подразумевает изменений со стороны пользователя.

CatID — это краеугольный камень всей идеи UCS и его нельзя менять на произвольную информацию, поэтому ее нужно брать исключительно из таблицы Категорий/Подкатегорий UCS.

Выполнение этого требования гарантирует, что, например, присваивая имя треку в котором присутствует законченное музыкальное произведение или песня вы соглашаетесь с тем, что

кратким сокращением (CatID) для этой пары Категории MUSIC и Подкатегории SONG будет MUSCSong

Другому пользователю системы UCS такой подход поможет понять, что за тип трека находится перед ним и какое у него содержимое.

К тому же, при использовании скриптов из современных программ-поисковиков аудио треков (типа Soundminer, BaseHead и др.) можно анализировать блок CatID и автоматически заносить информацию в поля Категорий/Подкатегорий в самой программе. А в некоторых случаях даже автоматически создавать описания треков, исходя из сочетания CatID с другими данными из имени файла.

TrackName - это следующий блок данных в имени файла UCS. Также, его можно использовать в качестве названия трека. Основная задача этого блока состоит в том, чтобы дать краткое описание содержимого трека, поэтому для облегчения визуального восприятия это поле обычно занимает около 25 символов.

Этот блок не предназначен для замены более информативного поля "Description" в метаданных трека, а скорее нужен для того, чтобы пользователь с первого взгляда понял, что представляет из себя трек, не прослушивая его.

CreatorID – содержит информацию о том кто записал, создал или издал трек. Стандарт UCS не регламентирует длину этого поля, поэтому в нем можно размещать и полные имена, однако для улучшения читаемости рекомендуется использовать аббревиатуры или инициалы, например Alexey Sergeev -> ASerg или Anton Ivanov -> Aliv

Желательно, чтобы автор выбрал свое имя (или сокращение от своего имени) и использовал его во всех своих треках. Это делается для того, чтобы автоматические скрипты поисковиков могли находить или сортировать треки по имени автора. Если же автор будет периодически менять свое имя, это может привести к определенной неразберихе в результатах выборки, когда разные имена в названии треков будут указывать на одного и того же автора.

SourceID - содержит название (или аббревиатуру) альбома, проекта или библиотеки автора к которому принадлежит данный трек. Опять же, стандарт UCS не регламентирует длину этого поля, поэтому в нем можно размещать и полные имена, однако рекомендуется использовать аббревиатуру или сокращение, если название альбома или библиотеки очень длинное.

В имени файла также можно использовать несколько необязательных блоков информации, которые позволяют еще больше улучшить информативность имени файла по стандарту UCS.

UserCategory следует после CatID и отделяется знаком дефиса «-» и обычно применяется пользователями для уточнения Категории/Подкатегории UCS. Этот блок опционален и не имеет жесткой привязки к структуре категорий UCS, как в случае с CatID и предусмотрен в первую очередь для нужд пользователей, в качестве дополнения Категории/Подкатегории. Пользователь сам вправе решать, какие данные заносить в этот блок. Обычно коммерческие библиотеки не используют этот блок информации, однако никаких жестких правил тут нет. Можно поместить туда основной стиль (жанр) музыкального трека, который содержится в данном файле. А для лучшего визуального отделения от следующего за ним названия трека рекомендуется оформлять блок UserCategory большими буквами. Например: EPIC, ROCK, HIPHOP, TECHNO.

MUSCSong-EPIC_Massive Drums_AGrin_CinematicStrikes.wav

VendorCategory – Этот блок похож на UserCategory, но отличается тем, что им часто пользуются производители библиотек для уточнения категорий ВНУТРИ своей библиотеки. Однако никто не запрещает пользоваться этим блоком для потребностей вашей базы данных музыкальных треков.

Блок VendorCategory отделяется от следующего за ним названия трека (TrackName) знаком дефиса «-».

MUSCSong_EPIC-Massive Drums_AGrin_CinematicStrikes.wav

UserData -Это последний блок данных в структуре имен файлов UCS, и он полностью опционален. Каждый пользователь или создатель библиотеки сам решает, как использовать эту информацию, поскольку она не присваивается какому-либо стандартному полю метаданных в программах, типа Soundminer, BaseHead или Soundly. В этом блоке можно хранить любую другую информацию, которую нужно передать конечному пользователю базы данных, например уникальный идентификатор внутри базы данных

MUSCSong_EPIC-Massive Drums_AGrin_CinematicStrikes_CB15RT456.wav

Примеры использования имен файлов формата UCS

Давайте рассмотрим несколько примеров имен файлов по стандарту UCS:

Пример 1: Законченный трек в стиле “Hip-Hop”, который имеет название “Fuel My Fire”, за авторством “MC Grizzy”, из альбома “Thug Life”. В имени файла используется блок VendorCategory в котором записан жанр HIPHOP

Примерное название трека с использованием UCS может иметь следующий вид:

MUSCSong_HIPHOP-Fuel My Fire_MCGrizzy_ThugLife.wav

Пример 2: Стем или соло-партия хора/ансамбля вокалистов из трека в стиле “Epic”, который имеет название “Elf Song”, за авторством “Igor Tikhonov”, из альбома “Magic Forest”. В имени файла используется блок VendorCategory в котором записан жанр EPIC

Примерное название трека с использованием UCS может иметь следующий вид:

MUSCChor_EPIC-Elf Song_IgorTikhonov_MagicForest.wav

Пример 3: Луп из трека в стиле “Techno”, который имеет название “Brain Damage”, за авторством “DJ Demon”, из библиотеки “Central Industrial”. В имени файла используется блок VendorCategory в котором записан жанр TECHNO

Примерное название трека с использованием UCS может иметь следующий вид:

MUSCLoop_TECHNO-Brain Damage_DJDemon_CentralIndustrial.wav

Пример 4: Партия струнных из трека в стиле “Trailer”, который имеет название “Intensity”, за авторством “Igor Vovin”, из альбома “Impact Elements”. В имени файла используется блок UserCategory в котором записан жанр TRAILER

Примерное название трека с использованием UCS может иметь следующий вид:

MUSCStr-TRAILER_Intensity_IgorVovin_ImpactElements.wav

Пример 5: Партия синтезаторов из трека в стиле “Industrial”, который имеет название “Steel Dawn”, за авторством группы “Blast”, из альбома “Dead Cities”. В имени файла используется блок VendorCategory в котором записан жанр INDUSTRIAL

Категория “Синтезаторы” (Synths) не прописана в Категориях/Подкатегориях UCS, поэтому в данном случае в качестве CatID мы имеем право использовать MUSCInst_ как обобщенную категорию инструментов, которые не подпадают под другие Подкатегории инструментов таблицы UCS.

Примерное название трека с использованием UCS может иметь следующий вид:

MUSCInst_INDUSTRIAL- Steel Dawn_Blast_DeadCities.wav.

Условно процесс каталогизации можно разделить на несколько этапов:

1. Чистка. Это всего лишь рекомендация для тех, кто начинает с уровня “файлопомойки” и нужна только в том случае, если вы хотите переделать все с нуля и упорядочить по-максимуму. Для начала нам нужно пройти по всем каталогам и бегло их почистить, выбросить все лишнее, что вы не слушаете и вряд ли будете. Нужно разделить все на треки и унифицировать форматы. Для нарезки и перекодирования есть масса программ, например Medieval CUE Splitter.
2. Распознавание и сопоставление. Это самое важное. Выполняем распознавание и сортировку композиций. Для этого нам нужна будет программа MusicBrainz Picard. Она доступна на всех основных ОС. Программа выполняет 3 основных роли: переименовывает файлы, заполняет теги и добавляет обложку. Правда обложки берутся с Archive.org, который сейчас заблокирован в РФ, поэтому без дополнительных телодвижений их придется добавлять вручную или другими программами. MusicBrainz содержит достаточно подробные данные о музыке, но к сожалению, данные о жанрах у него или отсутствуют или неадекватны. На этом поприще лидером является Last.fm, который как самостоятельный сервис давно стал неинтересен, но накопил много данных об исполнителях, жанрах, стилях и связях на их основе. После смены API многие скрипты перестали работать, а для MusicBrainz Picard остался только Last.fm.ng Для него главное проверить разделитель мульти-тега жанра. Справа отображаются опознанные треки, сгруппированные в альбомы. Но иногда нужно помогать программе распознавать треки. Файл может не опознаться, или в «кандидаты» на трек альбома может попасть несколько файлов. В этом случае придется выбирать вручную. Уровень совпадения обозначается цветом, от зеленого до красного. При проверке сравниваются некоторые теги и «fingerprint» трека.
3. Альбом помечается как золотой (желтый), если в нем присутствуют все треки, и нет никаких множественных интерпретаций. Наша цель - добиться максимального количества золотых альбомов. Также обратите внимание, что содержимое альбома может меняться в зависимости от выпускающей студии, иногда ручное изменение редакции поможет добиться более точного соответствия. Обработанные альбомы периодически сохраняйте в папку назначения. Music branes может распознать очень многое, но не все. В таком случае помогает плагин flobex 2000, чтобы распознать оставшиеся альбомы. Далее нужно сделать обложку.

В общем, да, процесс нетривиален, и при большой коллекции может быть длительным; кнопки «Сделать все как надо» здесь нет. Но, во-первых, в любом случае, правила, вкусы и предпочтения у каждого разные, соответственно, и настройки тоже у каждого будут разные. Во-вторых, на протяжении всего процесса нам помогали мощные программы, которые в значительной степени автоматизировали работу. Ну, и в-третьих, не обязательно с точностью повторять прохождение всех этапов; думаю, каждый может найти для себя оптимальный путь. Зато, в итоге мы получили:

- Единую упорядоченную структуру на уровне файловой системы
- ID3-теги, полностью заполненные (вплоть до композитора) по единым правилам
- Возможность сортировки по жанрам и другим тегам
- Обложки ко всем альбомам

- Эстетическое удовольствие от осознания факта, что у вас практически идеальный порядок.