Music Web API Documentation

Release Number 1.1.1

Published: 1/4/2013 2:34 PM

Gracenote, Inc.
2000 Powell Street, Suite 1500
Emeryville, California
94608-1804
www.gracenote.com

Music Web API Product Overview

Introduction

This document provides a technical overview of the main features of the Gracenote Music Web API.

About Gracenote

A pioneer in the digital media industry, Gracenote combines information, technology, services and applications to create ingenious entertainment solutions for the global market.

From media management, enrichment and discovery products to content identification technologies, Gracenote allows providers of digital media products and the content community to make their offerings more powerful and intuitive, enabling superior consumer experiences. The company's solutions integrate the broadest, deepest, and highest quality global metadata and enriched content with an infrastructure that services billions of searches a month from thousands of products used by hundreds of millions of consumers. Gracenote's customers include the biggest names in the consumer electronics, mobile, automotive, software and Internet industries. The company's partners in the entertainment community include major music publishers and labels, prominent independents and movie studios.

Gracenote technologies are used by leading online media services such as Apple iTunes, Spotify and Sony Music Unlimited, by innovative mobile apps such as Soundtracking and Path, and by leading consumer electronics manufacturers such as Sony, Philips and Acer.

For more information about Gracenote, please visit: www.gracenote.com.

What is the Music Web API?

The Music Web API is a Web Service optimized for lightweight client applications to retrieve and explore music-related information. Using this service, applications can deliver a compelling digital entertainment experience by giving users tools to manage and enjoy music collections on media devices, including desktop and mobile devices.

The Music Web API is the most comprehensive identification solution in the industry with the ability to recognize, categorize and organize any music source, including CDs, digital files, or audio streams.

Music Web API also seamlessly integrates with Gracenote's suite of products and provides the foundation for advanced services such as enriched content and linking to commerce.

Media recognition using Music Web API makes it possible for applications to access a variety of rich data available from Gracenote. After media has been recognized, applications can request and utilize:

- Album, track, and artist names
- Genre, origin, era and type descriptors
- Mood and tempo descriptors
- Music Enrichment content, including cover art, biographies, reviews and lyrics

System Requirements

The Music Web API does not have a binary SDK, Therefore, there are no particular platform requirements other than:

- Network Connection
- Ability to parse XML
- HTTPS

If your application requires recognizing music files using audio fingerprints, you will also need to use GNSDK APIs to create these fingerprints.

Music Data Elements

Music data available through the Music Web API is grouped into albums and tracks. An application can perform several types of queries to get album and track data:

- **Album search**: Returns metadata for one or more matching albums based on a text string search. Text string must one or more of the following: album title, artist title, or track title.
- **Album fingerprint**: Returns metadata for one or more matching albums based on a track's audio fingerprint.
- **Album TOC**: Returns metadata for a matching album based on an audio disk Table of Contents (TOC).
- **Album fetch**: Returns metadata for one or more matching albums based on a specific track, or metadata for a specific album.

Album Data

Value	Description
ALBUM	A matched album or albums.
ARTIST	The album artist's name.
DATE	The release year of the album.
GENRE	The album's genre, represented in the selected query language, if available. Also returns a NUM attribute for each genre, regardless of the language. Client applications can use this information to organize their local databases by genres.
MATCHED_TRACK_NUM	The number of the matched track for queries searching for a specific track in one or more albums.

TITLE	The album's title
TRACK	Nodes for each track returned. See Track Data Model.
TRACK_ COUNT	The number of tracks in the album.
URL	The URL for the album cover art. The TYPE attribute's value is either COVERART (for this album) or GENRE_COVERART (cover art representing the album's genre. The SIZE attribute indicates the cover art size. The client application should use the returned URLs immediately and never store them for later use. There is no guarantee that the URL is valid for more than a short period of time after it is returned. The format of the URL is subject to change over time. The client application should not modify or interpret the URL.

Track Data

Value	Description
TRACK	Node for individual tracks of an album.
ARTIST	Returned if the track's artist is different from the album's artist. For example, many compilation albums will list the album-level artist as Various Artists and the performers of the individual tracks in track level artist.
ARTIST_ ERA	The time era of the artist.
ARTIST_ ORIGIN	The artist's place of origin.
ARTIST_ TYPE	The kind of artist.
GENRE	Returned if the track's genre is different from the album's genre. If the track's genre is missing, the track inherits the album's genre.
MOOD	The mood of the track
TEMPO	The tempo of the track
TITLE	The track's title.
TRACK_ NUM	The track's number in the album.

Music Web API Implementation Guide

About the Gracenote Music Web API

This document presents common implementation guidelines for music identification and exploration using the Gracenote Music Web API.

The Gracenote Music Web API is an XML-based HTTP protocol that provides simple, yet highly functional access Gracenote Media Recognition Service. Using the protocol, client applications send queries and receive responses as XML through HTTP. You can use these interfaces to get detailed Gracenote metadata about albums and audio tracks.

Queries and Responses

The Gracenote Web API is an XML-based protocol: both query and response messages carry payloads in the form of XML documents. The root element for all queries is <QUERIES> containing one <QUERY> element. All responses have a root element of type <RESPONSES> containing one <RESPONSE> element.
Example Queries Element

```
<QUERIES>
    <QUERY>
    ...
    </QUERY>
</QUERIES>
```

Example Responses Element

```
<RESPONSES>
    <RESPONSE>
    ...
    </RESPONSE>
</RESPONSES>
```

Supported HTTPS Protocols

Client applications request information by issuing XML queries to the Gracenote Media Recognition Service and receive back XML responses. The Gracenote Media Recognition Service uses HTTPS for secure communication. All transmissions use UTF-8 encoding and the HTTPS Secure Socket Layer (SSL) cryptographic protocol.

Web API client applications must submit all Web API queries using an HTTP POST request. After establishing a secure connection, client applications can retrieve metadata (images, biographies, reviews, and others) using the HTTP GET.

Contact Gracenote Professional Services for the Gracenote Media Recognition Service URL to use for your client application.

HTTP Compression

To save on bandwidth, standard HTTP compression will be supported on the servers. If the client chooses to use compression, gzip is the recommended compression algorithm. The client can invoke this through the standard method of enabling HTTP compression (specifying the "Accept-Encoding: gzip" line in the request's header).

Posting Queries and Responses

You can send XML queries directly to the Gracenote Media Recognition Service from your browser. This is useful to test the service and learn how to use the Gracenote Web APIs. Two popular browser plug-ins for Firefox are Poster and RESTclient . To use these plug-ins, enter the Gracenote Media Recognition Service URL and an XML query. You must include your Gracenote client ID string and User ID string in the AUTH block of each query. See Registration and Authentication.

Registration and Authentication

Gracenote Media Recognition Service authenticates all queries using a combination of two customer-specific identifiers: a Client ID string and a User ID string.

Currently, the Client ID string has the following format: 123456-789123456789012312. The first part is the Client ID, and the second part is the Client ID Tag.

Gracenote Professional Services provides the Client ID string as part of your licensing agreement.

Every query a client application sends to Gracenote must include both of Client ID string and User ID string inside an authorization <AUTH> block.

Example Query with authorization

```
<QUERIES>
    <AUTH>
        <CLIENT>client_id_string</CLIENT>
        <USER>user_id_string</USER>
    </AUTH>
    <QUERY>
    ...
    </QUERY>
</QUERIES>
```

Registering a Device

To obtain a User ID for a device, the client application must submit a REGISTER query. This query requires a valid Client ID. The client application needs to register a device only once. If registration succeeds, Gracenote returns a successful registration response that contains the User ID. Your application

should store the User ID in a non-volatile location on the device itself, and use it for authentication in all subsequent queries.

> Store the User ID in a location accessible only by your application. Users and third-party applications must be blocked from accessing or viewing the User ID unless it is encrypted or obscured in some way.

All subsequent non-registration queries must include an <AUTH> block containing both the Client ID string and the User ID string.

Example of REGISTER query

```
<QUERIES>
    <QUERY CMD="REGISTER">
        <CLIENT>client_id_string</CLIENT>
    </QUERY>
</QUERIES>
```

Example of a Successful Registration response

```
<RESPONSES>
    <RESPONSE STATUS="OK">
        <USER>user_id_string</USER>
    </RESPONSE>
<RESPONSES>
```

If user registration fails, the <RESPONSE> element will be empty and will carry the STATUS attribute ERROR. A separate <MESSAGE> element is included along with the response, containing a descriptive error message. For example, the following is a response for a registration query that was missing a valid client ID-Client ID Tag pair.
Example Registration error response

```
<RESPONSES>
    <MESSAGE>Missing or bad CLIENT information.</MESSAGE>
    <RESPONSE STATUS="ERROR">
    </RESPONSE>
</RESPONSES>
```

Setting the Language Preference

A query can include an optional <LANG> element to specify a preferred natural language in which to receive response metadata. The body of the <LANG> element should be a three-character language code as defined by the ISO 639-2 standard.

When possible, the Gracenote Media Recognition Service returns localized metadata in the language requested. If no metadata is available in the requested language, or if there is no language preference, metadata is returned in English (eng) by default.

Example of Language Preference

```
<QUERIES>
    <AUTH>
    ...
    </AUTH>
    <LANG>ger</LANG>
    <QUERY>
    ...
    </QUERY>
</QUERIES>
```

The following table lists the currently supported languages for the Music Web API

Language	Code
ARABIC	ara
BAHASA_INDONESIA	ind
BULGARIAN	bul
CHINESE_SIMP	qtb
CHINESE_TRAD	qtd
CROATIAN	scr
CZECH	cze
DANISH	dan
DUTCH	dut
ENGLISH	eng
FARSI	per
FINNISH	fin
FRENCH	fre
GERMAN	ger
GREEK	gre
HUNGARIAN	hun
ITALIAN	ita
JAPANESE	jpn
KOREAN	kor
NORWEGIAN	nor

POLISH	pol
PORTUGUESE_BRAZIL	por
ROMANIAN	rum
RUSSIAN	rus
SERBIAN	scc
SLOVAK	slo
SPANISH	spa
SWEDISH	swe
THAI	tha
TURKISH	tur
VIETNAMESE	vie

Application and SDK Info in Queries

<APP_INFO> and <SDK_INFO> are optional query elements, however Gracenote strongly encourages you to use them. When used, they should contain information sent by both the application and the SDK used (if any), respectively. Information in these two fields must be presented as a comma-separated list of key=value pairs and the value should be double-quoted. Suggested keys are:

app	Application name and version
sdk	SDK name and version
mfg	Manufacturer of device
os	Platform and operating system

Applications and SDKs should populate as many of these fields as possible, and are free to add additional fields, to aid debugging and information gathering.

Status Reporting

Each <RESPONSE> element carries a STATUS attribute indicating whether the corresponding query succeeded or failed. If a query fails because of an error of some kind, the <RESPONSE> element will have a STATUS attribute of ERROR and will be accompanied by a separate <MESSAGE> element describing the error.

Response Status	Description

OK	Query succeeded and produced at least one match
NO_MATCH	Query succeeded but did not produce a match
ERROR	Query failed

Example of Success Response
If the value of the STATUS attribute is OK, the body of the <RESPONSE> element contains the data returned in response to the query.

```
<RESPONSES>
    <RESPONSE STATUS="OK">
        <\!-\- ...Response data... \-->
    </RESPONSE>
</RESPONSES>
```

Example of No Match Response
If STATUS is NO_MATCH, the element body is empty.

```
<RESPONSES>
    <RESPONSE STATUS="NO_MATCH">
    </RESPONSE>
</RESPONSES>
```

Example of Failure response
The following example shows the response to a VIDEODISCSET_TOC query that was missing the TOC data to look up. The application should display the message string to the user, to allow for diagnosis of problems in the field; this information is required for Gracenote support representatives to help in solving both developer and end-user issues.

```
<RESPONSES>
    <MESSAGE>QUERY missing TOC information</MESSAGE>
        <RESPONSE STATUS="ERROR">
    </RESPONSE>
</RESPONSES>
```

General Gracenote Web API Best Practices

The following are best practices for designing client applications with the Gracenote Web API.

Retries

- When an error occurs, your client application should retry once. If it still gets an error, give up and notify the user. The client application must not keep retrying. Doing so can potentially damage the Gracenote Media Recognition Service.

Error Strings

- You should ensure that error strings from the Web API are easy to retrieve. Preferably, you should display this string to the end user, or at least optionally display it. Error strings are contained in the <MESSAGE> element of a response.
- Do not try to interpret the error string, rely on it being in a certain format, or try to re-format or otherwise modify it. Error strings can and will change without notice.

Handling XML Responses

- When dealing with the XML responses, understand that new elements and new attributes will be added to extend the protocol.
- The client application should ignore any XML elements and attributes that are not relevant.

Missing Elements

- Design your client code to handle missing elements gracefully. Even elements that always should be there (such as GN_ID) may be missing due to a bug. Should this happen, the client application should not crash or seriously malfunction.

Ordered Elements

- When dealing with ordered elements (RANK or ORD attributes), the client application must be tolerant of gaps in the enumeration, as well as objects of unknown type that appear in the list.
- The client application should ignore objects of unknown type and skip to the one that has the next higher ordinal or rank and is of a known type.

Working with GN_IDs

- Your client application should treat all Gracenote identifiers (GN_IDs) as opaque strings. Do not try to interpret or parse them. Caching them locally for client application use is permitted.

Queuing Requests

- For most standard device configurations, Gracenote has typically recommended a maximum of 5 concurrent threads per core simultaneously querying via the Gracenote Web Services API. This assumes that the application properly handles conditions such as timeouts, retries, and error handling.

Handling Timeouts

- In the event of a timeout when querying the Gracenote Web Services API, Gracenote recommends one immediate retry. This applies to all non-registration Web Services API queries.

Music Web API Query Types

Each query has a query type denoting the type of operation requested. You specify the query type using the CMD attribute attached to the <QUERY> element.
Query names follow a standard pattern:

<Target_Object>_FETCH	A Fetch query uses a Gracenote ID (GN_ID) to locate and return an item.

<Target_Object>_ SEARCH	A Search query uses a text string to search for and return an item.
<Target_Object>_ LOOKUP	A Lookup query uses a Table of Contents (TOC) or fingerprint (FP) to locate and return the item

Where <Target_Object> is the object being queried.

Query Types

The Gracenote Music Web API provides the following query types. For details and examples of these queries, see Music Web API Reference. For information about the data model, see Music Web API Data Model Guide.

ALBUM_ SEARCH	Returns metadata for one or more matching albums based on a text string search. Text string must one or more of the following: album title, artist title, or track title.
ALBUM_FIN-GERPRINT	Returns metadata for one or more matching albums based on a fingerprint derived from an audio file or ambient audio recording.
ALBUM_TOC	Returns metadata for a matching album based on an audio disk Table of Contents (TOC).
ALBUM_ FETCH	Returns metadata for one or more matching albums based on a track GN_ID, or one album for an album GN_ID.

By default, these queries return album matches without cover art. Using modes and options, you can refine or extend the response.

The Lyrics Web API provides the following query types:

LYRIC_ SEARCH	The lyric search query looks up lyrics in Gracenote database, based on text inputs. The search returns one or more matching lyrics, containing an excerpt of the lyric lines, along with the GN_IDs of the matched lyrics.
LYRIC_ FETCH	The lyric fetch query returns the full lyric text based on the GN_ID of the matched lyric. The GN_ID is usually obtained from a prior lyric search.

The Inventory Web API provides the INVENTORY query type.

INVENTORY	You can use this query mechanism to build and maintain databases out of Gracenote data. Using Inventory, you can acquire a list of all available data objects of a certain type, as well as deltas, which are periodic incremental changes. This is useful for managing lyrics data.

Audio File Fingerprinting

An Audio fingerprint is data that uniquely identifies an audio track. The Gracenote Media Recognition Service uses audio fingerprints to match the audio from a client application to the Gracenote Music Database.

You can generate audio fingerprints using the GNSDK API. Refer to the GNSDK documentation for more information.

Types of Fingerprints

There are two basic types of fingerprints: MusicID-File and MusicID-Stream. MusicID-File recognizes audio files such as those generated when ripping a CD. It uses the first portion of the audio file for recognition. MusicID-Stream can recognize a snippet if audio that can be taken from anywhere within the track making it suitable for identifying streamed audio (such as an internet stream) or ambient audio (such as that recorded from a microphone).

Gracenote audio fingerprints are XML documents. The Music Web API can process fingerprints in compressed or uncompressed formats.

Your application can retain fingerprints for a collection of audio files so they can be used later in queries. For example, your application can fingerprint an entire collection of files in a background thread and reuse them later.

Using a MusicID-File Fingerprint

MusicID-File fingerprints use the first 15 seconds of detectable audio from an audio stream. Any silence at the beginning of the track is ignored. Because of this, Gracenote recommends the client application process at least the first 30 seconds of the audio stream to account for silence at the start of the track

There can be edge cases in which there is more than 30 seconds of silence at the beginning. In such cases, you can still create a fingerprint if there is at least 15 seconds of detectable audio in the track.

Example MusicID-File fingerprint

```
<GNFPX_FP_QUERY VERSION="1.0">
 <ALGORITHM>
  <NAME>Cantametrix</NAME>
  <VERSION>1.1</VERSION>
  <QUERY>Cantametrix"1.1</QUERY>
 </ALGORITHM>
 <FP_BLOCKS>
  <FP_BLOCK TYPE="SIGNED_INT_16" COUNT="30" ENCODING="ASCII" SEPARATOR=" ">1005
1242 2315 4588</FP_BLOCK>
 </FP_BLOCKS>
</GNFPX_FP_QUERY>
```

To use a MusicID-File fingerprint in the Music Web API, include only the FP_Block portion of the fingerprint XML document in the query.

```
<QUERY CMD="ALBUM_FINGERPRINT">
 <FINGERPRINT ALGORITHM="a_fingertprint_algorithm" VERSION="a_version_number">
  <DATA>an_xml_data_block</DATA>
 </FINGERPRINT>
</QUERY>
```

Using a MusicID-Stream Fingerprint

The MusicID-Stream fingerprint uses approximately 6.5 seconds of audio stream for most use cases. However, additional time may be required.

To use a MusicID-Stream fingerprint, include the entire fingerprint XML document as *entity encoded (XML-escaped)* data.

```
<QUERIES>
  <AUTH>
    <CLIENT>client_id_string</CLIENT>
    <USER>user_id_string</USER>
  </AUTH>
  <QUERY CMD="ALBUM_FINGERPRINT">
    <FINGERPRINT ALGORITHM="a_fingertprint_algorithm" VERSION="a_ver-
sion_number">
      <DATA>an_xml_data_block</DATA>
    </FINGERPRINT>
  </QUERY>
</QUERIES>
```

Best Practices for Recognizing Audio from the Microphone

The following practices are recommended when recognizing audio from the microphone:

- Provide clear and concise onscreen instructions on how to record the audio:
  - Most failed recognitions are due to incorrect operation by the user. Provide clear and concise instructions to help the user correctly operate the application to result in a higher match rate and a better user experience.
- While recording audio from the microphone display a progress indicator:
  - When recording from the microphone, the application can receive status updates. The status updates indicate what percentage of the recording is completed. Use this information to display a progress bar (indicator) to notify the user. When recording has finished use vibration or a tone (or both) to notify the user.
- Visual only notifications can hamper the user experience because:
  - The user may not see notifications if they are holding the recording device up to an audio source. Also, the user may pull the device away from an audio source to check if recording has completed. This may result in a poor quality recording.
    In general, your application should display an animation to indicate the application is performing an operation. If the application appears to stop, the user may believe the application has crashed.
  - If no match is found, redisplay the user instructions and ask the user to try again.

Music Web API Use Cases and Common Queries

Introduction

The following use cases and common queries are based on the following Music Web API definitions:

ALBUM_SEARCH	Returns metadata for one or more matching albums based on a text string search. Text string must one or more of the following: album title, artist title, or track title.
ALBUM_FINGERPRINT	Returns metadata for one or more matching albums based on a track's audio fingerprint.
ALBUM_TOC	Returns metadata for a matching album based on an audio disk table of contents (TOC).
ALBUM_FETCH	Returns metadata for one or more matching albums based on a track GN_ID, or one album for an album GN_ID.

By default, these queries return album matches without cover art. Using modes and options, you can refine or extend the response.

Use Cases

Explore Album and Tracks on Audio CD (using TOC)

1. User inserts an audio CD into a device.
2. Client application reads the CD TOC and sends the ALBUM_TOC query to Gracenote.
3. Gracenote returns metadata for the matching album (assuming TOCs mostly unique).

Explore Album(s) containing an Audio File (using Fingerprint)

1. Client application presents list of local (on device) Track files to user.
2. User chooses Track to explore. If no fingerprint for the track is available locally, client application generates a fingerprint using GNSDK APIs.
3. User chooses whether to see best match album match or all matches. Assuming best match choice for this example.
4. Client applicationsends Track fingerprint to Gracenote using ALBUM_FINGERPRINT as SINGLE_BEST.
5. Gracenote returns metadata for best matching album that includes the audio file.

Explore Album(s) using Text Search

1. User enters text strings for one or more of: artist name, album name, track name.
2. User chooses whether to see best match album or all matches. Assuming best match choice for this example.
3. Client application sends ALBUM_SEARCH query using this data as SINGLE_BEST.
4. Gracenote returns metadata for best matching album for the text data provided.

Explore an Album using an Album GN_ID

1. Client application presents a list of albums to the user after performing a multiple album query using ALBUM_FINGERPRINT or ALBUM_SEARCH.
2. User selects specific album.
3. Client application sends the Album GN_ID to GN using ALBUM_FETCH.
4. Gracenote returns metadata for the specific album, including all its track metadata.

Explore a Specific Track using Text Search

1. User enters text strings for artist name, album name, and track name.
2. Client application sends ALBUM_SEARCH query using this data.
3. Assuming there is a match, Gracenote returns metadata for the specific track, and its containing album.

Explore a Specific Track using a Track GN_ID

1. Client application presents a list of tracks to the user after performing an album query using ALBUM_FINGERPRINT, ALBUM_TOC, or ALBUM_SEARCH.
2. User selects a specific track.
3. Client application sends the Track GN_ID to GN using ALBUM_FETCH.
4. Gracenote returns metadata for the specific track, and its containing album.

Common Queries

Getting a Single Album Match

ALBUM_FETCH with an Album GN_ID

Query	ALBUM_FETCH
Mode	Any or None
Options	Any or None
Input	Album GN_ID
Returns	Metadata for the one matching album without cover art.

ALBUM_TOC with an Audio Disk TOC

Query	ALBUM_TOC
Mode	Any or None
Options	Any or None
Input	Album TOC
Returns	Metadata for the one matching album without cover art.

ALBUM_FINGERPRINT with a Track Fingerprint

Without Cover Art (SINGLE_BEST=YES)

Query	ALBUM_FINGERPRINT
Mode	SINGLE_BEST=YES
Options	Any or None
Input	Track Fingerprint
Returns	Metadata for the one matching album without cover art.

With Cover Art (SINGLE_BEST_COVER=YES)

Query	ALBUM_FINGERPRINT
Mode	SINGLE_BEST_COVER=YES
Options	Any or None
Input	Track Fingerprint
Returns	Metadata for the one matching album with cover art.

With Cover Art (SINGLE_BEST=YES and SELECT_EXTENDED=COVER)

Query	ALBUM_FINGERPRINT
Mode	SINGLE_BEST=YES
Options	SELECT_EXTENDED=COVER
Input	Track Fingerprint
Returns	Metadata for the one matching album with cover art.

ALBUM_FETCH with a Track GN_ID

Without Cover Art (SINGLE_BEST=YES)

Query	ALBUM_FETCH
Mode	SINGLE_BEST=YES
Options	Any or None
Input	Track GN_ID
Returns	Metadata for the one matching album without cover art.

With Cover Art (SINGLE_BEST_COVER=YES) or (SINGLE_BEST=YES and SELECT_EXTENDED=COVER)

Query	ALBUM_FETCH
Mode	SINGLE_BEST=YES
Options	SELECT_EXTENDED=COVER
Input	Track GN_ID
Returns	Metadata for the one matching album with cover art.

ALBUM_SEARCH with a Text String

Without Cover Art (SINGLE_BEST=YES)

Query	ALBUM_SEARCH
Mode	SINGLE_BEST=YES
Options	Any or None
Input	Text string: One or more of the following: album title, artist title, or track title.
Returns	Metadata for the one matching album without cover art.

With Cover Art (SINGLE_BEST_COVER=YES) or (SINGLE_BEST=YES and SELECT_EXTENDED=COVER)

Query	ALBUM_SEARCH
Mode	SINGLE_BEST=YES

Options	SELECT_EXTENDED=COVER
Input	Text string: One or more of the following: album title, artist title, or track title.
Returns	Metadata for the one matching album with cover art.

Getting Multiple Album Matches

ALBUM_FINGERPRINT with a Track Fingerprint

Without Cover Art

Query	ALBUM_FINGERPRINT
Mode	Any or None
Options	Any or None
Input	Track Fingerprint.
Returns	Metadata for one or more matching albums without cover art.

With Cover Art (SELECT_EXTENDED=COVER)

Query	ALBUM_FINGERPRINT
Mode	Any or None
Options	SELECT_EXTENDED=COVER
Input	Track Fingerprint.
Returns	Metadata for one or more matching albums with cover art.

ALBUM_FETCH with a Track GN_ID

Without Cover Art

Query	ALBUM_FETCH
Mode	Any or None
Options	Any or None
Input	Track GN_ID.
Returns	Metadata for one or more matching albums without cover art.

With Cover Art (SELECT_EXTENDED=COVER)

Query	ALBUM_FETCH
Mode	Any or None
Options	SELECT_EXTENDED=COVER
Input	Track GN_ID.
Returns	Metadata for one or more matching albums with cover art.

ALBUM_SEARCH with a Text String

Without Cover Art

Query	ALBUM_SEARCH
Mode	Any or None
Options	Any or None
Input	Text string: One or more of the following: album title, artist title, or track title.
Returns	Metadata for the one or more matching albums without cover art.

With Cover Art (SELECT_EXTENDED=COVER)

Query	ALBUM_SEARCH
Mode	Any or None
Options	SELECT_EXTENDED=COVER
Input	Text string: One or more of the following: album title, artist title, or track title.
Returns	Metadata for the one or more matching albums with cover art.

Extending Music Web API Query with Options

SELECT_EXTENDED Option

SELECT_ EXTENDED VALUE	Description

COVER	Returns the cover art. This is the preferred way to return cover art. It is the same as using the obsolete RETURN_COVER standalone option.
REVIEW	Returns URL(s) to album review(s).
ARTIST_ BIOGRAPHY	Returns URL(s) to artist biography(ies).
ARTIST_ IMAGE	Returns URL to artist image.
CONTENT	Returns URLs to cover art, reviews, artist biographies, and artist image. To return a subset of these, specify a comma-separated list of the corresponding values.

For example, the following two queries both return all available SELECT_EXTENDED metadata.

Query	ALBUM_FETCH
Mode	Any or None
Options	SELECT_EXTENDED=COVER, REVIEW, ARTIST_BIOGRAPHY, ARTIST_IMAGE
Input	Track or Album GN_ID
Returns	Metadata for one or more matching albums.

Or

Query	ALBUM_FETCH
Mode	Any or None
Options	CONTENT
Input	Track or Album GN_ID
Returns	Metadata for one or more matching albums.

SELECT_DETAIL Option

Using SELECT_DETAIL, you can specify the hierarchy levels to return for list types.

Label	Description
GENRE:3LEVEL	Returns up to three levels for the current genre hierarchy (DETAILED or SIMPLIFIED).
MOOD:2LEVEL	Returns up to two levels for the Mood hierarchy.
TEMPO:3LEVEL	Returns up to three levels for the Tempo hierarchy.

ARTIST_ORIG-IN:4LEVEL	Returns up to four levels for the Artist Origin hierarchy.
ARTIST_ERA:2LEVEL	Returns up to two levels for the Artist Era hierarchy.
ARTIST_ TYPE:2LEVEL	Returns up top two levels for the Artist Type hierarchy.

For example, the following query returns up to three levels of genre hierarchy, and up to four levels of artist origin for the provided GN_ID.

Query	ALBUM_FETCH
Mode	Any or None
Options	SELECT_DETAIL=GENRE:3LEVEL, ARTIST_ORIGIN:4LEVEL
Input	Track or Album GN_ID
Returns	Metadata for one or more matching albums.

COVER_SIZE Option

You can use COVER_SIZE to specify the order of cover art sizes in the response.

For example, the following query returns available cover art in this order: large, extra large, small, medium, and thumbnail.

Query	ALBUM_FETCH
Mode	Any or None
Options	COVER_SIZE=LARGE,XLARGE,SMALL,MEDIUM,THUMBNAIL
Input	Track or Album GN_ID
Returns	Metadata for one or more matching albums.

FALLBACK_GENRECOVER Option

You can use FALLBACK_GENRECOVER to ensure that an image will be returned for a query that requests cover art, even if the actual art is not available. In such cases, Gracenote returns a default image representing the album's genre.

Query	ALBUM_FETCH
Mode	Any or None
Options	FALLBACK_GENRECOVER=YES

Input	Track or Album GN_ID
Returns	Metadata for one or more matching albums.

Music Web API Data Model Guide

Introduction

This document describes the basic data model used by the Music Web API.

Album Data Model

Value	Sub-Value	Description
ALBUM	@ORD	Node for matched album(s). ORD indicates the order of the album in a response that returns more than one album.
ARTIST		The album artist's name as a text string.
DATE		The release year of the album.
GENRE	@NUM, @ID	The album's genre as a text string, represented in the selected query language, if available. If the genre string is not available in the client's preferred language, or if no language preference is set in the query, the value is in English. The NUM attribute contains a unique number for each genre, regardless of the language. Client applications can use this information to organize their local databases by genres.
GN_ID		Unique Gracenote identifier for the matched album.
MATCHED_TRACK_NUM		The number of the matched track for queries searching for a specific track in one or more albums.
RANGE		Node for queries that return multiple albums.
	COUNT	Indicates the total number of matched albums.
	START	Indicates the start position (based on the ORD album value) for the albums returned in the response.
	END	Indicates the end position (based on the ORD album value) for the albums returned in the response.
TITLE		The album's title as a text string.
TRACK		Nodes for each track returned. See Track Data Model.

TRACK_COUNT		The number of tracks in the album.
URL	@TYPE, @SIZE	The URL for the album cover art. The TYPE attribute's value is either COVERART (for this album) or GENRE_COVERART (cover art representing the album's genre. The SIZE attribute indicates the cover art size. The client application should use the returned URLs immediately and never store them for later use. There is no guarantee that the URL is valid for more than a short period of time after it is returned. The format of the URL is subject to change over time. The client application should not modify or interpret the URL.

Track Data Mode

Value	Sub-Value	Description
TRACK		Node for individual tracks.
ARTIST		Returned if the track's artist is different from the album's artist. For example, many compilation albums will list the album-level artist as Various Artists and the performers of the individual tracks in track level artist.
ARTIST_ERA	@ID	
ARTIST_ORIGIN	@ID	
ARTIST_TYPE	@ID	
GENRE		Returned if the track's genre is different from the album's genre. If the track's genre is missing, the track inherits the album's genre.
GENRE	@NUM, @ID	
GN_ID		Gracenote identifier for the track.
MOOD	@ID	
TEMPO	@ID	
TITLE		The track's title.
TRACK_NUM		The track's number.

Music Web API Reference

Music Web API Query Types

The Music Web API provides the following query types. By default, these queries return album matches without cover art. You can refine or extend the response to include cover art and other content using modes and options, described in SELECT_EXTENDED, SELECT_DETAIL, and Other Options. However, we currently only allow cover art to be fetched if the user has elected to return only the single best result. To learn about data elements and attributes used by the Music Web API, see Music Web API Data Model Guide.

ALBUM_SEARCH	Returns metadata for one or more matching albums based on a text string search. Text string must one or more of the following: album title, artist title, or track title.
ALBUM_FINGERPRINT	Returns metadata for one or more matching albums based on a track's audio fingerprint.
ALBUM_TOC	Returns metadata for a matching album based on an audio disk Table of Contents (TOC).
ALBUM_FETCH	Returns metadata for one or more matching albums based on a track GN_ID, or one album for an album GN_ID.

ALBUM_SEARCH

ALBUM_SEARCH returns metadata for one or more matching albums based on a text string search. The text string must be one or more of the following: artist name, album title, or track title.
By default, this query does not return cover art in the response. You can refine or extend the response using modes and options. To include cover art you can use SINGLE_BEST_COVER mode or SELECT_EXTENDED option with a value COVER. See Single-Best Mode with ALBUM_SEARCH.

You can also extend the response using options to include additional metadata, such as artist origin, artist era, and artist type, track mood, and track tempo. See SELECT_EXTENDED, SELECT_DETAIL, and Other Options.

If more than one album is returned, the order is determined by a combination of match closeness and album popularity. Availability of cover art does not affect the best match selection. An album's order is indicated by the ORD attribute in the response.

Parameters

Parameter	Description

TEXT TYPE="ARTIST"	Optional. The artist name. Returns the matching album or albums for the specified artist.
TEXT TYPE="ALBUM_TITLE"	Optional. The title of the album. Returns the matching album or albums with the specified name.
TEXT TYPE="TRACK_TITLE"	Optional. The track title. Returns the matching album or albums that contain the track, and the track's metadata. If the query does not specify a track title, it returns metadata for all tracks in the matching album or albums.
RANGE	Optional. Enables paging through results. Use RANGE with progressively higher START and END values. By default, a query returns up to 10 matching albums. See Paging Results with the RANGE Parameter Paging Results Using the RANGE Parameter.

Single-Best Mode with ALBUM_SEARCH

A single best album search returns the only one album that is the best match for your search criteria. Two modes are available for single-best matching: SINGLE_BEST and SINGLE_BEST_COVER. In general, use SINGLE_BEST to return the best album match without its cover art. To get the single best album and its cover art, use SINGLE_BEST_COVER.

Mode	Description
SINGLE_BEST	Optional. Returns album metadata, without cover art, based on the specified parameters (one or more of artist name, album title, and track title). Unless there is no match, this query always returns only one album (known as the best match). .
SINGLE_BEST_COVER	Optional. Returns album metadata, including cover art, based on artist, album, and track strings. Unless there is no match, this query always returns only one album (known as the best match), along with its cover art.

Options with ALBUM_SEARCH

There are several options you can use to refine or extend the response, including SELECT_EXTENDED and SELECT_DETAIL. For information about using these options, see SELECT_EXTENDED, SELECT_DETAIL, and Other Options.

Paging Results with the RANGE Parameter

Using the optional RANGE parameter in a query enables paging through results. To do this, make repeated queries and include RANGE with progressively higher START and END values.
By default, ALBUM_SEARCH returns up to 10 matching albums. If more than one album is returned, the

order is determined by a combination of match closeness and album popularity. An album's order is indicated by the ORD attribute in the response.

If the query used the RANGE parameter, the response will contain a RANGE element. This element includes a COUNT value that indicates the number of matching albums from the query overall. The START and END values in the response indicate which albums are being returned based on their ORD value.

In the example below, the query specifies Fountains of Wayne as the artist name and Stacy's Mom as the track title. The first query specifies a RANGE with START=1 and END=2.
This track was popular, so the results show it appeared in 135 albums (COUNT=135). Based on the query, the first two matching albums are returned. The client application can request the remaining matches by sending additional queries using RANGE with progressive values for START and END.

Example Query

```
<QUERIES>
  <AUTH>
    <CLIENT>client_id_string</CLIENT>
    <USER>user_id_string</USER>
  </AUTH>
  <LANG>eng</LANG>
  <COUNTRY>usa</COUNTRY>
  <QUERY CMD="ALBUM_SEARCH">
    <TEXT TYPE="ARTIST">Fountains of Wayne</TEXT>
    <TEXT TYPE="TRACK_TITLE">Stacy's Mom</TEXT>
    <RANGE>
      <START>1</START>
      <END>2</END>
    </RANGE>
  </QUERY>
</QUERIES>
```

Example Response

```
<RESPONSES>
  <RESPONSE STATUS="OK">
    <RANGE>
      <COUNT>135</COUNT>
      <START>1</START>
      <END>2</END>
    </RANGE>
    <ALBUM ORD="1">
      <GN_ID>a_gracenote_identifier</GN_ID>
      <ARTIST>Fountains Of Wayne</ARTIST>
      <TITLE>Welcome Interstate Managers</TITLE>
      <DATE>2003</DATE>
      <GENRE NUM="64661" ID="27057">Classic Soft & Pop Rock</GENRE>
      <MATCHED_TRACK_NUM>3</MATCHED_TRACK_NUM>
      <TRACK_COUNT>16</TRACK_COUNT>
```

```
      <TRACK>
        <TRACK_NUM>3</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <TITLE>Stacy's Mom</TITLE>
      </TRACK>
    </ALBUM>
    <ALBUM ORD="2">
      <GN_ID>a_gracenote_identifier</GN_ID>
      <ARTIST>Various Artists</ARTIST>
      <TITLE>Now That's What I Call Music\! Vol. 14</TITLE>
      <DATE>2003</DATE>
      <GENRE NUM="106214" ID="35980">Other Pop</GENRE>
      <MATCHED_TRACK_NUM>16</MATCHED_TRACK_NUM>
      <TRACK_COUNT>20</TRACK_COUNT>
      <TRACK>
        <TRACK_NUM>16</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>Fountains Of Wayne</ARTIST>
        <TITLE>Stacy's Mom</TITLE>
        <GENRE NUM="64661" ID="27057">Classic Soft & Pop Rock</GENRE>
      </TRACK>
    </ALBUM>
  </RESPONSE>
</RESPONSES>
```

Queries and Responses for ALBUM_SEARCH

Example Query using ARTIST, ALBUM_TITLE, and TRACK_TITLE

```
<QUERIES>
  <AUTH>
    <CLIENT>client_id_string</CLIENT>
    <USER>user_id_string</USER>
  </AUTH>
  <LANG>eng</LANG>
  <COUNTRY>usa</COUNTRY>
  <QUERY CMD="ALBUM_SEARCH">
    <MODE>SINGLE_BEST_COVER</MODE>
    <TEXT TYPE="ARTIST">The Beatles</TEXT>
    <TEXT TYPE="ALBUM_TITLE">Help</TEXT>
    <TEXT TYPE="TRACK_TITLE">The Night Before</TEXT>
  </QUERY>
</QUERIES>
```

Example Response using ARTIST, ALBUM_TITLE, and TRACK_TITLE

```
<RESPONSES>
  <RESPONSE STATUS="OK">
    <ALBUM>
      <GN_ID>a_gracenote_identifier</GN_ID>
      <ARTIST>The Beatles</ARTIST>
```

```
      <TITLE>Help\!</TITLE>
      <DATE>1965</DATE>
      <GENRE NUM="64660" ID="27056">60's Oldies</GENRE>
      <MATCHED_TRACK_NUM>2</MATCHED_TRACK_NUM>
      <TRACK_COUNT>14</TRACK_COUNT>
      <TRACK>
        <TRACK_NUM>2</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <TITLE>The Night Before</TITLE>
      </TRACK>
      <URL TYPE="COVERART" SIZE="MEDIUM">a_temporary_url</URL>
    </ALBUM>
  </RESPONSE>
</RESPONSES>
```

Example Query using ARTIST, ALBUM_TITLE, and SINGLE_BEST_COVER

```
<QUERIES>
  <AUTH>
    <CLIENT>client_id_string</CLIENT>
    <USER>user_id_string</USER>
  </AUTH>
  <LANG>eng</LANG>
  <COUNTRY>usa</COUNTRY>
  <QUERY CMD="ALBUM_SEARCH">
    <MODE>SINGLE_BEST_COVER</MODE>
    <TEXT TYPE="ARTIST">The Beatles</TEXT>
    <TEXT TYPE="ALBUM_TITLE">Help</TEXT>
  </QUERY>
</QUERIES>
```

Example Response using ARTIST, ALBUM_TITLE, and SINGLE_BEST_COVER

```
<RESPONSES>
  <RESPONSE STATUS="OK">
    <ALBUM>
      <GN_ID>a_gracenote_identifier</GN_ID>
      <ARTIST>The Beatles</ARTIST>
      <TITLE>Help\!</TITLE>
      <DATE>1965</DATE>
      <GENRE NUM="64660" ID="27056">60's Oldies</GENRE>
      <TRACK_COUNT>14</TRACK_COUNT>
      <TRACK>
        <TRACK_NUM>1</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <TITLE>Help\!</TITLE>
      </TRACK>
      <TRACK>
        <TRACK_NUM>2</TRACK_NUM>
```

```
        <GN_ID>a_gracenote_identifier</GN_ID>
        <TITLE>The Night Before</TITLE>
      </TRACK>
      ...
      <TRACK_NUM>14</TRACK_NUM>
      <GN_ID>a_gracenote_identifier</GN_ID>
      <TITLE>Dizzy Miss Lizzy</TITLE>
    </TRACK>
    <URL TYPE="COVERART" SIZE="MEDIUM">a_temporary_url</URL>
  </ALBUM>
</RESPONSE>
</RESPONSES>
```

ALBUM_FINGERPRINT

ALBUM_FINGERPRINT returns metadata for one or more matching albums based on a track's audio fingerprint.

You can generated audio fingerprints using the GNSDK API. Refer to the GNSDK documentation for more information.

By default, this query does not return cover art in the response. You can refine or extend the response using modes and options. To include cover art you can use SINGLE_BEST_COVER mode or SELECT_EXTENDED option with a value COVER. See Single-Best Modes with ALBUM_FINGERPRINT.

You can also extend the response using options to include additional metadata, such as artist origin, artist era, and artist type, track mood, and track tempo. See SELECT_EXTENDED, SELECT_DETAIL, and Other Options.

If more than one album is returned, the order is determined by a combination of match closeness and album popularity. Availability of cover art does not affect the best match selection. An album's order is indicated by the ORD attribute in the response.

Single-Best Modes with ALBUM_FINGERPRINT

A single best album search returns the only one album that is the best match for your search criteria. Two modes are available for single-best matching: SINGLE_BEST and SINGLE_BEST_COVER. In general, use SINGLE_BEST to return the best album match without its cover art. To get the single best album and its cover art, use SINGLE_BEST_COVER.

Mode	Description
SINGLE_BEST	Optional. Returns album metadata, without cover art, based on the specified parameters (one or more of artist name, album title, and track title). Unless there is no match, this query always returns only one album (known as the best match).
SINGLE_BEST_COVER	Optional. Returns album metadata, including cover art, based on artist, album, and track strings. Unless there is no match, this query always returns only one album (known as the best match), along with its cover art.

Options with ALBUM_FINGERPRINT

There are several options you can use to refine or extend the response, including SELECT_EXTENDED and SELECT_DETAIL. For information about using these options, see SELECT_EXTENDED, SELECT_DETAIL, and Other Options.

Queries and Responses for ALBUM_FINGERPRINT

Example Query using a MusicID-File Fingerprint

```
<QUERIES>
  <AUTH>
    <CLIENT>client_id_string</CLIENT>
    <USER>user_id_string</USER>
  </AUTH>
  <LANG>eng</LANG>
  <COUNTRY>usa</COUNTRY>
  <QUERY CMD="ALBUM_FINGERPRINT">
    <MODE>SINGLE_BEST_COVER</MODE>
    <FINGERPRINT ALGORITHM="a_fingertprint_algorithm" VERSION="a_ver-
sion_number">
      <DATA>an_xml_data_block</DATA>
    </FINGERPRINT>
    <OPTION>
      <PARAMETER>COVER_SIZE</PARAMETER>
      <VALUE>SMALL,MEDIUM,THUMBNAIL,LARGE,XLARGE</VALUE>
    </OPTION>
    <OPTION>
      <PARAMETER>FALLBACK_GENRECOVER</PARAMETER>
      <VALUE>YES</VALUE>
    </OPTION>
    <OPTION>
      <PARAMETER>SELECT_EXTENDED</PARAMETER>
      <VALUE>COVER,ARTIST_BIOGRAPHY,ARTIST_IMAGE</VALUE>
    </OPTION>
  </QUERY>
</QUERIES>
```

Example Response using a MusicID-File Fingerprint

```
<RESPONSES>
  <RESPONSE STATUS="OK">
    <ALBUM>
      <GN_ID>a_gracenote_identifier</GN_ID>
      <ARTIST>Queen</ARTIST>
      <TITLE>Greatest Hits I</TITLE>
      <DATE>1994</DATE>
      <GENRE NUM="106202" ID="35970">Hard Rock</GENRE>
      <MATCHED_TRACK_NUM>1</MATCHED_TRACK_NUM>
      <TRACK_COUNT>17</TRACK_COUNT>
      <TRACK>
```

```
        <TRACK_NUM>1</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <TITLE>Bohemian Rhapsody</TITLE>
      </TRACK>
      <URL TYPE="COVERART" SIZE="SMALL">a_temporary_url</URL>
      <URL TYPE="ARTIST_BIOGRAPHY">a_temporary_url</URL>
      <URL TYPE="ARTIST_IMAGE">a_temporary_url</URL>
    </ALBUM>
  </RESPONSE>
</RESPONSES>
```

Example Query using MusicID-Stream

```
<QUERIES>
  <AUTH>
    <CLIENT>client_id_string</CLIENT>
    <USER>user_id_string</USER>
  </AUTH>
  <QUERY CMD="ALBUM_FINGERPRINT">
    <FINGERPRINT ALGORITHM="a_fingertprint_algorithm" VERSION="a_ver-
sion_number">
      <DATA>an_xml_data_block</DATA>
    </FINGERPRINT>
  </QUERY>
</QUERIES>
```

Example Response using MusicID-Stream

```
<RESPONSES>
  <RESPONSE STATUS="OK">
    <ALBUM ORD="1">
      <GN_ID>a_gracenote_identifier</GN_ID>
      <ARTIST>Wilco</ARTIST>
      <TITLE>A Ghost is Born</TITLE>
      <DATE>2004</DATE>
      <GENRE NUM="105227" ID="35479">Alternative Roots</GENRE>
      <MATCHED_TRACK_NUM>5</MATCHED_TRACK_NUM>
      <TRACK_COUNT>12</TRACK_COUNT>
      <TRACK>
        <TRACK_NUM>5</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <TITLE>Hummingbird</TITLE>
      </TRACK>
    </ALBUM>
  </RESPONSE>
</RESPONSES>
```

ALBUM_TOC

Returns metadata for a matching album based on an audio disk Table of Contents (TOC). This query may return more than one matching album. However, most TOCs are unique, so it is likely that the response will be one album only. CDs with a few tracks could possibly have the same TOC, although this is rare. You can use a single best match mode to ensure one match. See Single-Best Modes with ALBUM_TOC.

By default, this query does not return cover art in the response. You can refine or extend the response using modes and options. To include cover art you can use SINGLE_BEST_COVER mode or SELECT_EXTENDED option with a value COVER. See Single-Best Modes with ALBUM_TOC.

You can also extend the response using options to include additional metadata, such as artist origin, artist era, and artist type, track mood, and track tempo. See SELECT_EXTENDED, SELECT_DETAIL, and Other Options.

Single-Best Modes with ALBUM_TOC

ALBUM_TOC usually returns a single matching album. This is because most TOCs are unique. However, to ensure a single response, you can use a single best mode to return the single best match. Two modes are available for single-best matching: SINGLE_BEST and SINGLE_BEST_COVER. In general, use SINGLE_BEST to return the best album match without its cover art. To get the single best album and its cover art, use SINGLE_BEST_COVER.

Mode	Description
SINGLE_BEST	Optional. Returns album metadata, without cover art, based on the specified parameters (one or more of artist name, album title, and track title). Unless there is no match, this query always returns only one album (known as the best match).
SINGLE_BEST_COVER	Optional. Returns album metadata, including cover art, based on artist, album, and track strings. Unless there is no match, this query always returns only one album (known as the best match), along with its cover art.

Options with ALBUM_TOC

There are several options you can use to refine or extend the response, including SELECT_EXTENDED and SELECT_DETAIL. For information about using these options, see SELECT_EXTENDED, SELECT_DETAIL, and Other Options in this document.

Example Queries and Responses for ALBUM_TOC

Example Query using SINGLE_BEST_COVER

```
<QUERIES>
  <AUTH>
    <CLIENT>client_id_string</CLIENT>
    <USER>user_id_string</USER>
  </AUTH>
```

```
  <LANG>eng</LANG>
  <COUNTRY>usa</COUNTRY>
  <QUERY CMD="ALBUM_TOC">
    <MODE>SINGLE_BEST_COVER</MODE>
    <TOC>
      <OFFSETS>150 20512 30837 50912 64107 78357 90537 110742 126817
144657 153490 160700 175270 186830 201800 218010 237282 244062 262600
272929
    </OFFSETS>
  </TOC>
</QUERY>
</QUERIES>
```

ALBUM_FETCH

ALBUM_FETCH returns metadata for only one album based on an album GN_ID as input. By default, this query does not return cover art in the response. You can refine or extend the response using modes and options. To include cover art you can use SELECT_EXTENDED option with a value COVER.

You can also extend the response using options to include additional metadata, such as artist origin, artist era, and artist type, track mood, and track tempo. See Options with ALBUM_FETCH.

ALBUM_FETCH using Track instead of Album GN_IDs

Most use cases for ALBUM_FETCH use an album ID as input. However, the query also accepts track GN_IDs as input. In this case, the query returns metadata for one or more matching albums that contain the track. The response indicates which track was matched in each album by returning a MATCHED_TRACK_NUM field.

If more than one album is returned, the order is determined by a combination of match closeness and album popularity. Availability of cover art does not affect the best match selection. An album's order is indicated by the ORD attribute in the response.

Options with ALBUM_FETCH

There are several options you can use to refine or extend the response, including SELECT_EXTENDED and SELECT_DETAIL. For information about using these options, see SELECT_EXTENDED, SELECT_DETAIL, and Other Options.

Example Queries and Responses for ALBUM_FETCH

Example Query

```
<QUERIES>  <AUTH>
    <CLIENT>client_id_string</CLIENT>
    <USER>user_id_string</USER>
  </AUTH>
  <LANG>eng</LANG>
  <COUNTRY>usa</COUNTRY>
  <QUERY CMD="ALBUM_FETCH">
```

```
    <GN_ID>a_gracenote_identifier</GN_ID>
  </QUERY>
</QUERIES>
```

Example Response

```
<RESPONSES>
  <RESPONSE STATUS="OK">
    <ALBUM>
      <GN_ID>a_gracenote_identifier</GN_ID>
      <ARTIST>The Beatles</ARTIST>
      <TITLE>Help\!</TITLE>
      <DATE>1965</DATE>
      <GENRE NUM="64660" ID="27056">60's Oldies</GENRE>
      <TRACK_COUNT>14</TRACK_COUNT>
      <TRACK>
        <TRACK_NUM>1</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <TITLE>Help\!</TITLE>
      </TRACK>
      …
      <TRACK>
        <TRACK_NUM>13</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <TITLE>Yesterday</TITLE>
      </TRACK>
      <TRACK>
        <TRACK_NUM>14</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <TITLE>Dizzy Miss Lizzy</TITLE>
      </TRACK>
    </ALBUM>
  </RESPONSE>
</RESPONSES>
```

SELECT_EXTENDED, SELECT_DETAIL, and Other Options

The Gracenote Music Web API provides several modes and options to tailor the query responses. The following tables show the options that apply to each query type.

Query and Option Summary

ALBUM_SEARCH

SINGLE_BEST and SINGLE_BEST_COVER Modes		COVER_SIZE Option	FALLBACK_GENRECOVER Option	RETURN_COVER Option	SELECT_EXTENDED Option	SELECT_DETAIL Option

Yes		Yes with SINGLE_BEST_COVER mode	Yes with SINGLE_BEST_COVER mode	No. Cover art is controlled SELECT_EXTENDED=COVER.	Yes	Yes

ALBUM_FINGERPRINT

SINGLE_BEST and SINGLE_BEST_COVER Modes		COVER_SIZE Option	FALLBACK_GENRECOVER Option	RETURN_COVER Option	SELECT_EXTENDED Option	SELECT_DETAIL Option
Yes		Yes with SINGLE_BEST_COVER mode.	Yes with SINGLE_BEST_COVER mode.	No. Cover art is controlled by SELECT_EXTENDED=COVER.	Yes	Yes

ALBUM_TOC

SINGLE_BEST and SINGLE_BEST_COVER Modes		COVER_SIZE Option	FALLBACK_GENRECOVER Option	RETURN_COVER Option	SELECT_EXTENDED Option	SELECT_DETAIL Option
Yes		Yes with SINGLE_BEST_COVER mode.	Yes with SINGLE_BEST_COVER mode.	No. Cover art is controlled SELECT_EXTENDED=COVER.	Yes	Yes

ALBUM_FETCH

SINGLE_BEST and SINGLE_BEST_COVER Modes		COVER_SIZE Option	FALLBACK_GENRECOVER Option	RETURN_COVER Option	SELECT_EXTENDED Option	SELECT_DETAIL Option
No.		Yes with SELECT_EXTENDED=COVER, or obsolete RETURN_COVER option enabled.	Yes with SELECT_EXTENDED=COVER, or obsolete RETURN_COVER option enabled.	Yes, but obsolete. Recommended method is SELECT_EXTENDED=COVER.	Yes	Yes

```
<QUERIES>
  <AUTH>
    <CLIENT>client_id_string</CLIENT>
    <USER>user_id_string</USER>
  </AUTH>
  <LANG>eng</LANG>
  <COUNTRY>usa</COUNTRY>
  <QUERY CMD="ALBUM_SEARCH">
    <MODE>SINGLE_BEST</MODE>
    <TEXT TYPE="ARTIST">The Beatles</TEXT>
    <OPTION>
      <PARAMETER>SELECT_EXTENDED</PARAMETER>
      <VALUE>COVER,LINK,ARTIST_BIOGRAPHY</VALUE>
    </OPTION>
    <OPTION>
      <PARAMETER>PREFER_XID</PARAMETER>
      <VALUE>applealbumid</VALUE>
    </OPTION>
  </QUERY>
</QUERIES>
```

```
<RESPONSES>
  <RESPONSE STATUS="OK">
    <ALBUM>
      <GN_ID>a_gracenote_identifier</GN_ID>
      <ARTIST>The Beatles</ARTIST>
      <TITLE>The Beatles (White Album) \[Disc 1\]</TITLE>
      <DATE>1968</DATE>
      <GENRE NUM="106204" ID="35972">Psychedelic</GENRE>
      <XID DATASOURCE="applealbumid">401126224</XID>
      <TRACK_COUNT>17</TRACK_COUNT>
      <TRACK>
        <TRACK_NUM>1</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <TITLE>Back In The USSR</TITLE>
        <GENRE NUM="64660" ID="27056">60's Oldies</GENRE>
        <XID DATASOURCE="appletrackid">401126225</XID>
        <XID DATASOURCE="gracenote" DATATYPE="isrc">GBAYE0601644</XID>
        <XID DATASOURCE="gracenote" DATATYPE="isrc">GBAYE0900726</XID>
        <URL TYPE="ARTIST_BIOGRAPHY">a_temporary_url</URL>
      </TRACK>
      <TRACK>
        <TRACK_NUM>2</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <TITLE>Dear Prudence</TITLE>
        <GENRE NUM="64660" ID="27056">60's Oldies</GENRE>
        <XID DATASOURCE="appletrackid">401126240</XID>
        <XID DATASOURCE="appletrackid">402094784</XID>
        <XID DATASOURCE="gracenote" DATATYPE="isrc">GBAYE0601645</XID>
        <XID DATASOURCE="gracenote" DATATYPE="isrc">GBAYE0900727</XID>
```

```
        <URL TYPE="ARTIST_BIOGRAPHY">a_temporary_url</URL>
      </TRACK>
      <TRACK>
        <TRACK_NUM>3</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <TITLE>Glass Onion</TITLE>
        <GENRE NUM="64660" ID="27056">60's Oldies</GENRE>
        <XID DATASOURCE="appletrackid">401126255</XID>
        <XID DATASOURCE="appletrackid">402094792</XID>
        <XID DATASOURCE="gracenote" DATATYPE="isrc">GBAYE0601646</XID>
        <XID DATASOURCE="gracenote" DATATYPE="isrc">GBAYE0900728</XID>
        <URL TYPE="ARTIST_BIOGRAPHY">a_temporary_url</URL>
      </TRACK>
      …
      <URL TYPE="COVERART" SIZE="MEDIUM">a_temporary_url</URL>
      <URL TYPE="ARTIST_BIOGRAPHY">a_temporary_url</URL>
    </ALBUM>
  </RESPONSE>
</RESPONSES>
```

COVER_SIZE Option

The COVER_SIZE option lets the client applications select an order of preference for the cover art size. It does not limit or filter the cover sizes returned by a query, just their order. Gracenote strongly encourages the use of the COVER_SIZE option.

Specify the order as a comma-separated list in the VALUE node. Supported sizes are XLARGE, LARGE, SMALL, MEDIUM, and THUMBNAIL. The Gracenote Media Recognition Service tries to obtain cover art in the first preferred size. If this is not available, it tries for the second size, and so on. Only specify the sizes you want to be ordered. By default, Gracenote Media Recognition Service returns images in this order: MEDIUM, LARGE, SMALL, XLARGE, and THUMBNAIL.

Example Query using COVER_SIZE

```
<QUERIES>
  <AUTH>
    <CLIENT>client_id_string</CLIENT>
    <USER>user_id_string</USER>
  </AUTH>
  <LANG>eng</LANG>
  <COUNTRY>usa</COUNTRY>
  <QUERY CMD="ALBUM_SEARCH">
    <MODE>SINGLE_BEST_COVER</MODE>
    <TEXT TYPE="ARTIST">smiths</TEXT>
    <TEXT TYPE="ALBUM_TITLE">singles</TEXT>
    <TEXT TYPE="TRACK_TITLE">How Soon is Now?</TEXT>
    <OPTION>
      <PARAMETER>COVER_SIZE</PARAMETER>
      <VALUE>LARGE,XLARGE,SMALL,MEDIUM,THUMBNAIL</VALUE>
    </OPTION>
```

```
  </QUERY>
</QUERIES>
```

Example Response using COVER_SIZE

```
<RESPONSES>
<RESPONSE STATUS="OK">
  <ALBUM>
    <GN_ID>a_gracenote_identifier</GN_ID>
    <ARTIST>The Smiths</ARTIST>
    <TITLE>Singles</TITLE>
    <DATE>1995</DATE>
    <GENRE NUM="106176" ID="35966">Indie</GENRE>
    <MATCHED_TRACK_NUM>6</MATCHED_TRACK_NUM>
    <TRACK_COUNT>18</TRACK_COUNT>
    <TRACK>
      <TRACK_NUM>6</TRACK_NUM>
      <GN_ID>a_gracenote_identifier</GN_ID>
      <TITLE>How Soon Is Now?</TITLE>
    </TRACK>
    <URL TYPE="COVERART" SIZE="LARGE">a_temporary_url</URL>
  </ALBUM>
</RESPONSE>
</RESPONSES>
```

FALLBACK_GENRECOVER Option (for Genre Cover Art)

The FALLBACK_GENRECOVER option instructs the Gracenote Media Recognition Service to return the album's genre cover art if the actual cover art is not available.
By default, this option is disabled. If the response returns genre cover art the URL attribute TYPE is set to GENRE_COVERART.

```
<URL TYPE="GENRE_COVERART" SIZE="SMALL">...</URL>
```

Example Query

```
<QUERIES>
  <AUTH>
    <CLIENT>client_id_string</CLIENT>
    <USER>user_id_string</USER>
  </AUTH>
  <LANG>eng</LANG>
  <COUNTRY>usa</COUNTRY>
  <QUERY CMD="ALBUM_SEARCH">
    <MODE>SINGLE_BEST_COVER</MODE>
    <TEXT TYPE="ARTIST">smiths</TEXT>
```

```
    <TEXT TYPE="ALBUM_TITLE">singles</TEXT>
    <TEXT TYPE="TRACK_TITLE">How Soon is Now?</TEXT>
    <OPTION>
      <PARAMETER>FALLBACK_GENRECOVER</PARAMETER>
      <VALUE>YES</VALUE>
    </OPTION>
  </QUERY>
</QUERIES>
```

SELECT_EXTENDED Option

Using SELECT_EXTENDED option, you can extend the responses of any query type.

SELECT_ EXTENDED VALUE	Description
COVER	Returns the cover art. This is the preferred way to return cover art. It is the same as using the obsolete RETURN_COVER standalone option.
REVIEW	Returns URL(s) to album review(s).
ARTIST_ BIOGRAPHY	Returns URL(s) to artist biography or biographies.
ARTIST_ IMAGE	Returns URL to artist image.
CONTENT	Returns URLs to cover art, reviews, artist biographies, and artist image. To return a subset of these, specify a comma-separated list of the corresponding values.

Example Query using COVER,ARTIST_IMAGE

```
<QUERIES>
  <AUTH>
    <CLIENT>client_id_string</CLIENT>
    <USER>user_id_string</USER>
  </AUTH>
  <LANG>eng</LANG>
  <COUNTRY>usa</COUNTRY>
  <QUERY CMD="ALBUM_FETCH">
    <GN_ID>a_gracenote_identifier</GN_ID>
    <OPTION>
      <PARAMETER>SELECT_EXTENDED</PARAMETER>
      <VALUE>COVER,ARTIST_IMAGE</VALUE>
    </OPTION>
  </QUERY>
</QUERIES>
```

Example Response using COVER,ARTIST_IMAGE

```
<RESPONSES>
  <RESPONSE STATUS="OK">
    <ALBUM>
      <GN_ID>a_gracenote_identifier</GN_ID>
      <ARTIST>Talking Heads</ARTIST>
      <TITLE>Stop Making Sense</TITLE>
      <DATE>1984</DATE>
      <GENRE NUM="75234" ID="31388">New Wave</GENRE>
      <MATCHED_TRACK_NUM>7</MATCHED_TRACK_NUM>
      <TRACK_COUNT>9</TRACK_COUNT>
      <TRACK>
        <TRACK_NUM>7</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <TITLE>What A Day That Was</TITLE>
      </TRACK>
      <URL TYPE="COVERART" SIZE="SMALL">url_to_get_data</URL>
    </ALBUM>
  </RESPONSE>
</RESPONSES>
```

Example Query using CONTENT

```
<QUERIES>
  <AUTH>
    <CLIENT>client_id_string</CLIENT>
    <USER>user_id_string</USER>
  </AUTH>
  <LANG>eng</LANG>
  <COUNTRY>usa</COUNTRY>
  <QUERY CMD="ALBUM_FETCH">
    <GN_ID>a_gracenote_identifier</GN_ID>
    <OPTION>
      <PARAMETER>SELECT_EXTENDED</PARAMETER>
      <VALUE>CONTENT</VALUE>
    </OPTION>
  </QUERY>
</QUERIES>
```

Example Response using CONTENT

```
<RESPONSES>
  <RESPONSE STATUS="OK">
    <ALBUM>
      <GN_ID>a_gracenote_identifier</GN_ID>
      <ARTIST>Talking Heads</ARTIST>
      <TITLE>Stop Making Sense</TITLE>
      <DATE>1984</DATE>
```

```
      <GENRE NUM="75234" ID="31388">New Wave</GENRE>
      <MATCHED_TRACK_NUM>7</MATCHED_TRACK_NUM>
      <TRACK_COUNT>9</TRACK_COUNT>
      <TRACK>
        <TRACK_NUM>7</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <TITLE>What A Day That Was</TITLE>
      </TRACK>
    </ALBUM>
  </RESPONSE>
</RESPONSES>
```

Fetching Artist Origin, Era, and Type

By default, metadata is not returned for artist origin, artist era, and artist type (OET). To get this meta-data, use the SELECT_EXTENDED option with the value ARTIST_OET. For era, the primary era is returned.

Example Query using ARTIST_OET

```
<QUERIES>
  <AUTH>
    <CLIENT>client_id_string</CLIENT>
    <USER>user_id_string</USER>
  </AUTH>
  <LANG>eng</LANG>
  <COUNTRY>usa</COUNTRY>
  <QUERY CMD="ALBUM_FETCH">
    <GN_ID>a_gracenote_identifier</GN_ID>
    <OPTION>
      <PARAMETER>RETURN_COVER</PARAMETER>
      <VALUE>YES</VALUE>
    </OPTION>
    <OPTION>
      <PARAMETER>SELECT_EXTENDED</PARAMETER>
      <VALUE>COVER,ARTIST_BIOGRAPHY,REVIEW,ARTIST_OET</VALUE>
    </OPTION>
  </QUERY>
</QUERIES>
```

Example Response using ARTIST_OET

```
<RESPONSES>
  <RESPONSE STATUS="OK">
    <ALBUM>
      <GN_ID>a_gracenote_identifier</GN_ID>
      <ARTIST>The Beatles</ARTIST>
      <ARTIST_ORIGIN ID="29967">United Kingdom</ARTIST_ORIGIN>
      <ARTIST_TYPE ID="29422">Male</ARTIST_TYPE>
```

```
      <ARTIST_ERA ID="29487">1960's</ARTIST_ERA>
      <TITLE>Help\!</TITLE>
      <DATE>1965</DATE>
      <GENRE NUM="64660" ID="27056">60's Oldies</GENRE>
      <TRACK_COUNT>14</TRACK_COUNT>
      <TRACK>
        <TRACK_NUM>1</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <TITLE>Help\!</TITLE>
        <URL TYPE="ARTIST_BIOGRAPHY">a_temporary_url</URL>
      </TRACK>
      …
    </ALBUM>
  </RESPONSE>
</RESPONSES>
```

Fetching Mood and Tempo

The Gracenote Service provides two metadata fields that describe the sonic attributes of an audio track: mood and tempo. These track-level descriptors capture the unique characteristics of a specific recording.

Mood is a perceptual descriptor of a piece of music, using emotional terminology that a typical listener might use to describe the audio track. Tempo is a description of the overall perceived speed or pace of the music. The Gracenote mood and tempo descriptor systems include hierarchical categories of increasing granularity, from very broad parent categories to more specific child categories.
The Music Web API can return track level mood descriptors for an album. The client can display mood descriptors c to the end user, or can use them to categorize music in the user's collection for organization, navigation, or playlist generation.

- Your application must be entitled to retrieve mood data. Contact Gracenote Professional Services for details.

To get mood and tempo data, use the SELECT_EXTENDED option in your query with keywords MOOD and TEMPO.

- You can use the SELECT_DETAIL option to define the level of granularity for mood. For mood, the Music Web API can deliver either DEFAULT (single level) or 2LEVEL for (two levels). For tempo, the Music Web API delivers a single level only.

Example Query using MOOD and TEMPO

```
<QUERIES>
  <AUTH>
    <CLIENT>client_id_string</CLIENT>
    <USER>user_id_string</USER>
  </AUTH>
  <LANG>eng</LANG>
  <COUNTRY>usa</COUNTRY>
  <QUERY CMD="ALBUM_FETCH">
```

```
    <GN_ID>a_gracenote_identifier</GN_ID>
    <OPTION>
      <PARAMETER>SELECT_EXTENDED</PARAMETER>
      <VALUE>MOOD,TEMPO</VALUE>
    </OPTION>
  </QUERY>
</QUERIES>
```

Example Response using MOOD and TEMPO

```
<RESPONSES>
  <RESPONSE STATUS="OK">
    <ALBUM>
      <GN_ID>a_gracenote_identifier</GN_ID>
      <ARTIST>Various Artists</ARTIST>
      <TITLE>O Brother, Where Art Thou?</TITLE>
      <DATE>2000</DATE>
      <GENRE NUM="64773" ID="27131">Original Film/TV Music</GENRE>
      <TRACK_COUNT>19</TRACK_COUNT>
      <TRACK>
        <TRACK_NUM>1</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>James Carter & The Prisoners</ARTIST>
        <TITLE>Po Lazuras</TITLE>
        <GENRE NUM="106221" ID="35985">Traditional Folk</GENRE>
        <MOOD ID="42966">Other</MOOD>
        <TEMPO ID="34303">40s</TEMPO>
      </TRACK>
      <TRACK>
        <TRACK_NUM>2</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>Harry McClintock</ARTIST>
        <TITLE>Big Rock Candy Mountain</TITLE>
        <GENRE NUM="106221" ID="35985">Traditional Folk</GENRE>
        <MOOD ID="42946">Easygoing</MOOD>
        <TEMPO ID="34318">90s</TEMPO>
      </TRACK>
      <TRACK>
        <TRACK_NUM>3</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>Norman Blake</ARTIST>
        <TITLE>You Are My Sunshine</TITLE>
        <GENRE NUM="75252" ID="31402">Bluegrass</GENRE>
        <MOOD ID="65323">Romantic</MOOD>
        <TEMPO ID="34300">30s</TEMPO>
      </TRACK>
      <TRACK>
        <TRACK_NUM>4</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
```

```
        <ARTIST>Alison Krauss</ARTIST>
        <TITLE>Down To The River To Pray</TITLE>
        <GENRE NUM="75252" ID="31402">Bluegrass</GENRE>
        <MOOD ID="65324">Sentimental</MOOD>
        <TEMPO ID="34307">50s</TEMPO>
      </TRACK>
      <TRACK>
        <TRACK_NUM>5</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>The Soggy Bottom Boys</ARTIST>
        <TITLE>I Am A Man Of Constant Sorrow</TITLE>
        <GENRE NUM="106221" ID="35985">Traditional Folk</GENRE>
        <MOOD ID="65327">Gritty</MOOD>
        <TEMPO ID="34316">80s</TEMPO>
      </TRACK>
      <TRACK>
        <TRACK_NUM>6</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>Chris Thomas King</ARTIST>
        <TITLE>Hard Time Killing Floor Blues</TITLE>
        <GENRE NUM="106226" ID="35990">Acoustic Blues</GENRE>
        <MOOD ID="42949">Melancholy</MOOD>
        <TEMPO ID="34318">90s</TEMPO>
      </TRACK>
      <TRACK>
        <TRACK_NUM>7</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>John Hartford</ARTIST>
        <TITLE>I Am A Man Of Constant Sorrow</TITLE>
        <GENRE NUM="75252" ID="31402">Bluegrass</GENRE>
        <MOOD ID="42946">Easygoing</MOOD>
        <TEMPO ID="34318">90s</TEMPO>
      </TRACK>
      <TRACK>
        <TRACK_NUM>8</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>The Whites</ARTIST>
        <TITLE>Keep On The Sunny Side</TITLE>
        <GENRE NUM="106225" ID="35989">Classic Country</GENRE>
        <MOOD ID="42946">Easygoing</MOOD>
        <TEMPO ID="34303">40s</TEMPO>
      </TRACK>
      <TRACK>
        <TRACK_NUM>9</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>Alison Krauss & Gillian Welch</ARTIST>
        <TITLE>I'll Fly Away</TITLE>
        <GENRE NUM="75252" ID="31402">Bluegrass</GENRE>
        <MOOD ID="42946">Easygoing</MOOD>
        <TEMPO ID="34307">50s</TEMPO>
```

```
      </TRACK>
      <TRACK>
        <TRACK_NUM>10</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>Emmylou Harris, Alison Krauss & Gillian Welch</ARTIST>
        <TITLE>Didn't Leave Nobody But The Baby</TITLE>
        <GENRE NUM="106221" ID="35985">Traditional Folk</GENRE>
        <MOOD ID="65327">Gritty</MOOD>
        <TEMPO ID="34307">50s</TEMPO>
      </TRACK>
      <TRACK>
        <TRACK_NUM>11</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>The Peasall Sisters</ARTIST>
        <TITLE>In The Highways</TITLE>
        <GENRE NUM="106221" ID="35985">Traditional Folk</GENRE>
        <MOOD ID="42946">Easygoing</MOOD>
        <TEMPO ID="34323">110s</TEMPO>
      </TRACK>
      <TRACK>
        <TRACK_NUM>12</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>The Cox Family</ARTIST>
        <TITLE>I Am Weary (Let Me Rest)</TITLE>
        <GENRE NUM="106221" ID="35985">Traditional Folk</GENRE>
        <MOOD ID="65323">Romantic</MOOD>
        <TEMPO ID="34307">50s</TEMPO>
      </TRACK>
      <TRACK>
        <TRACK_NUM>13</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>John Hartford</ARTIST>
        <TITLE>I Am A Man Of Constant Sorrow</TITLE>
        <GENRE NUM="75252" ID="31402">Bluegrass</GENRE>
        <MOOD ID="42942">Tender</MOOD>
        <TEMPO ID="34313">70s</TEMPO>
      </TRACK>
    </ALBUM>
  </RESPONSE>
</RESPONSES>
```

SELECT_DETAIL Option

Using the SELECT_DETAIL, you can specify how many levels of a hierarchy list to return. You can specify levels for genre, mood, and tempo. You can also choose levels for artist origin, artist era, and artist type, collectively known as OET values. When the query returns more than one value, the returned fields are ordered by hierarchy level using an ORD attribute.

Label	Description

GENRE:3LEVEL	Returns up to three levels for the current genre hierarchy (DETAILED or SIMPLIFIED).
MOOD:2LEVEL	Returns up to two levels for the Mood hierarchy.
TEMPO:3LEVEL	Returns up to three levels for the Tempo hierarchy.
ARTIST_ORIG-IN:4LEVEL	Returns up to four levels for the Artist Origin hierarchy.
ARTIST_ERA:2LEVEL	Returns up to two levels for the Artist Era hierarchy.
ARTIST_ TYPE:2LEVEL	Returns up top two levels for the Artist Type hierarchy.

When using SELECT_DETAIL, you can get up to the number of levels specified, but not necessarily all the levels. For example, depending on the artist, ARTIST_ORIGIN:4LEVEL returns one, two, three, or four levels. It depends on the data available for that artist.

It is not possible to establish parent-child relationships among the returned values. Gracenote strongly discourages this.

Example Query using SELECT_DETAIL

```
<QUERIES>
  <AUTH>
    <CLIENT>client_id_string</CLIENT>
    <USER>user_id_string</USER>
  </AUTH>
  <LANG>eng</LANG>
  <COUNTRY>usa</COUNTRY>
  <QUERY CMD="ALBUM_FETCH">
    <GN_ID>a_gracenote_identifier</GN_ID>
    <OPTION>
      <PARAMETER>SELECT_EXTENDED</PARAMETER>
      <VALUE>ARTIST_OET,MOOD,TEMPO</VALUE>
    </OPTION>
    <OPTION>
      <PARAMETER>SELECT_DETAIL</PARAMETER>
      <VALUE>GENRE:3LEVEL,MOOD:2LEVEL,TEMPO:3LEVEL,ARTIST_ORIG-
IN:4LEVEL,ARTIST_ERA:2LEVEL,ARTIST_TYPE:2LEVEL</VALUE>
    </OPTION>
  </QUERY>
</QUERIES>
```

Example Response using SELECT_DETAIL

```
<RESPONSES>
  <RESPONSE STATUS="OK">
    <ALBUM>
      <GN_ID>a_gracenote_identifier</GN_ID>
      <ARTIST>Various Artists</ARTIST>
      <TITLE>O Brother, Where Art Thou?</TITLE>
      <DATE>2000</DATE>
      <GENRE ORD="1" NUM="64645" ID="27045">Other</GENRE>
      <GENRE ORD="2" NUM="64773" ID="27131">Original Film/TV
Music</GENRE>
      <GENRE ORD="3" NUM="106275" ID="36029">General Film Music</GENRE>
      <TRACK_COUNT>19</TRACK_COUNT>
      <TRACK>
        <TRACK_NUM>1</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>James Carter & The Prisoners</ARTIST>
        <ARTIST_ORIGIN ORD="1" ID="29889">North America</ARTIST_ORIGIN>
        <ARTIST_ORIGIN ORD="2" ID="29908">United States</ARTIST_ORIGIN>
        <ARTIST_TYPE ORD="1" ID="29422">Male</ARTIST_TYPE>
        <ARTIST_TYPE ORD="2" ID="29436">Male Group</ARTIST_TYPE>
        <ARTIST_ERA ORD="1" ID="29488">1950's</ARTIST_ERA>
        <ARTIST_ERA ORD="2" ID="29513">Late 50's</ARTIST_ERA>
        <TITLE>Po Lazuras</TITLE>
        <GENRE ORD="1" NUM="64643" ID="27043">Roots</GENRE>
        <GENRE ORD="2" NUM="106221" ID="35985">Traditional Folk</GENRE>
        <GENRE ORD="3" NUM="65605" ID="27525">General Traditional
Folk</GENRE>
        <MOOD ORD="1" ID="42966">Other</MOOD>
        <TEMPO ORD="1" ID="34282">Slow Tempo</TEMPO>
        <TEMPO ORD="2" ID="34288">Slow</TEMPO>
        <TEMPO ORD="3" ID="34303">40s</TEMPO>
      </TRACK>
      <TRACK>
        <TRACK_NUM>2</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>Harry McClintock</ARTIST>
        <ARTIST_ORIGIN ORD="1" ID="29889">North America</ARTIST_ORIGIN>
        <ARTIST_ORIGIN ORD="2" ID="29908">United States</ARTIST_ORIGIN>
        <ARTIST_TYPE ORD="1" ID="29422">Male</ARTIST_TYPE>
        <ARTIST_TYPE ORD="2" ID="29426">Male</ARTIST_TYPE>
        <ARTIST_ERA ORD="1" ID="29491">Early 20th C.</ARTIST_ERA>
        <ARTIST_ERA ORD="2" ID="29525">Late 20's</ARTIST_ERA>
        <TITLE>Big Rock Candy Mountain</TITLE>
        <GENRE ORD="1" NUM="64643" ID="27043">Roots</GENRE>
        <GENRE ORD="2" NUM="106221" ID="35985">Traditional Folk</GENRE>
        <GENRE ORD="3" NUM="65605" ID="27525">General Traditional
Folk</GENRE>
        <MOOD ORD="1" ID="42946">Easygoing</MOOD>
        <MOOD ORD="2" ID="65339">Charming / Easygoing</MOOD>
        <TEMPO ORD="1" ID="34283">Medium Tempo</TEMPO>
```

```
        <TEMPO ORD="2" ID="34291">Medium Fast</TEMPO>
        <TEMPO ORD="3" ID="34318">90s</TEMPO>
      </TRACK>
      <TRACK>
        <TRACK_NUM>3</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>Norman Blake</ARTIST>
        <ARTIST_ORIGIN ORD="1" ID="29889">North America</ARTIST_ORIGIN>
        <ARTIST_ORIGIN ORD="2" ID="29908">United States</ARTIST_ORIGIN>
        <ARTIST_ORIGIN ORD="3" ID="30209">Tennessee</ARTIST_ORIGIN>
        <ARTIST_TYPE ORD="1" ID="29422">Male</ARTIST_TYPE>
        <ARTIST_TYPE ORD="2" ID="29426">Male</ARTIST_TYPE>
        <ARTIST_ERA ORD="1" ID="29486">1970's</ARTIST_ERA>
        <TITLE>You Are My Sunshine</TITLE>
        <GENRE ORD="1" NUM="64643" ID="27043">Roots</GENRE>
        <GENRE ORD="2" NUM="75252" ID="31402">Bluegrass</GENRE>
        <GENRE ORD="3" NUM="65574" ID="27516">New Traditional Blue-
grass</GENRE>
        <MOOD ORD="1" ID="65323">Romantic</MOOD>
        <MOOD ORD="2" ID="42982">Sweet / Sincere</MOOD>
        <TEMPO ORD="1" ID="34282">Slow Tempo</TEMPO>
        <TEMPO ORD="2" ID="34287">Very Slow</TEMPO>
        <TEMPO ORD="3" ID="34300">30s</TEMPO>
      </TRACK>
      ...
    </ALBUM>
  </RESPONSE>
</RESPONSES>
```

Advanced Topics

Specifying a Country-Specific Genre Hierarchy

The Music Web API supports an optional COUNTRY code for queries. If the specified country has a custom genre hierarchy, the Gracenote Media Recognition Service returns the custom hierarchy instead of the default hierarchy.

Currently, Japan, Korea, and Taiwan, all have their own hierarchies, and all European countries use the same custom EU hierarchy. This list will evolve over time.

The Music Web API uses ISO 3166-1 alpha-3 country codes (http://en.wikipedia.org/wiki/ISO_3166-1_alpha-3). Although not required, your client application should send the COUNTRY code in lower case.

Example Query using COUNTRY to Specify a Genre Hierarchy

```
<QUERIES>
  <AUTH>
    <CLIENT>client_id_string</CLIENT>
    <USER>user_id_string</USER>
  </AUTH>
  <LANG>eng</LANG>
```

```
  <COUNTRY>jpn</COUNTRY>
  <QUERY CMD="ALBUM_FETCH">
    <GN_ID>a_gracenote_identifier</GN_ID>
    <OPTION>
      <PARAMETER>SELECT_EXTENDED</PARAMETER>
      <VALUE>COVER</VALUE>
    </OPTION>
  </QUERY>
</QUERIES>
```

Example Response using COUNTRY to specify a Genre Hierarchy

The response indicates the genre hierarchy for Japan. Notice that many tracks for this album are Western Pop. For example, the "global" genre for track 1, The Prisoner, is Traditional Folk. However, in the Japanese genre hierarchy, the genre for track 1 is Western Pop.

```
<RESPONSES>
  <RESPONSE STATUS="OK">
    <ALBUM>
      <GN_ID>a_gracenote_identifier</GN_ID>
      <ARTIST>Various Artists</ARTIST>
      <TITLE>O Brother, Where Art Thou?</TITLE>
      <DATE>2000</DATE>
      <GENRE NUM="105712" ID="35786">Soundtracks</GENRE>
      <TRACK_COUNT>19</TRACK_COUNT>
      <TRACK>
        <TRACK_NUM>1</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>James Carter & The Prisoners</ARTIST>
        <TITLE>Po Lazuras</TITLE>
        <GENRE NUM="105675" ID="35773">Western Pop</GENRE>
      </TRACK>
      <TRACK>
        <TRACK_NUM>2</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>Harry McClintock</ARTIST>
        <TITLE>Big Rock Candy Mountain</TITLE>
        <GENRE NUM="105675" ID="35773">Western Pop</GENRE>
      </TRACK>
      <TRACK>
        <TRACK_NUM>3</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>Norman Blake</ARTIST>
        <TITLE>You Are My Sunshine</TITLE>
        <GENRE NUM="105675" ID="35773">Western Pop</GENRE>
      </TRACK>
      <TRACK>
        <TRACK_NUM>4</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>Alison Krauss</ARTIST>
        <TITLE>Down To The River To Pray</TITLE>
```

```
        <GENRE NUM="105675" ID="35773">Western Pop</GENRE>
      </TRACK>
      <TRACK>
        <TRACK_NUM>5</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>The Soggy Bottom Boys</ARTIST>
        <TITLE>I Am A Man Of Constant Sorrow</TITLE>
        <GENRE NUM="105675" ID="35773">Western Pop</GENRE>
      </TRACK>
      <TRACK>
        <TRACK_NUM>6</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>Chris Thomas King</ARTIST>
        <TITLE>Hard Time Killing Floor Blues</TITLE>
        <GENRE NUM="105715" ID="35789">Blues</GENRE>
      </TRACK>
      <TRACK>
        <TRACK_NUM>7</TRACK_NUM>
        <GN_ID>a_gracenote_identifier</GN_ID>
        <ARTIST>John Hartford</ARTIST>
        <TITLE>I Am A Man Of Constant Sorrow</TITLE>
        <GENRE NUM="105675" ID="35773">Western Pop</GENRE>
      </TRACK>
      ...
      <URL TYPE="COVERART" SIZE="MEDIUM">a_temporary_url</URL>
    </ALBUM>
  </RESPONSE>
</RESPONSES>
```