GNSDK for ODP: Release Notes

Version 3.07.7

Published: July 14, 2015

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

Table of Contents

About the GNSDK for ODP Package

The GNSDK for The Open Developer Program is packaged as a .zip file. The package includes the following:

- builds: Contains the make files to build the samples for each of the supported platforms.
- docs: Contains the GNSDK documentation.
- images: Contains standard JPG files to display when playing tracks that were not matched by GNSDK.
- include: Contains global and library-specific headers.
- lib: Contains DLLs and shared libraries.
- lib_static: Contains static libraries.
- samples: Contains all files necessary to build and run each library's sample application.
- wrappers: Contains the following object-oriented wrapper implementation with samples to ease application development. These have not been tested extensively.
    - C++
    - C#
    - Java
    - python

Products Included

The SDK includes the following products:

- Gracenote MusicID (CD TOC, Text, Fingerprint recognition for audio)
- Gracenote MusicID-File (Audio-file-based recognition)
- Gracenote Link (Content and third-party data retrieval)
- Gracenote Playlist (Automated Playlist generation and MoodGrid support)
- Gracenote MusicID Match
- Gracenote Video

System Requirements

This release supports the following platforms (OS and Architecture):

OS	CPU Architecture
Microsoft Windows	x86 and x64
Linux	x86, x64, MIPS-32EL, ARM (32 bit) and ARM HF (32 bit)
Mac OS X	x86 and x64

This release also includes libraries for the following platforms, but they have not been tested extensively:

OS	CPU Architecture
iOS	x86, x64, ARMv7 and ARMv7s (32 bit)
Android	ARM (32 bit), ARMEABI, ARMEABI-v7a

Gracenote Client ID String and License File

Please contact Gracenote Global Services & Solutions team for your Client ID String and License File. These are both required when initializing a GNSDK application.

How to Run the Samples

To help you learn how to implement GNSDK features, Gracenote provides a working, command-line based sample application for each library, the “main.c” file. This file uses only ANSI C and C Standard library functions. Although you may find running the application to be helpful, it is probably more beneficial to step through its execution with a debugger to observe its internal library usage. Please see README_Samples.txt for more details.

Current Limitations

This release has the following known limitation:

- While it is best practice for applications to release all generated handles, it is intended that gnsdk_manager_shutdown() will clean up by releasing any handles that have not been released. However, gnsdk_manager_shutdown() does not currently release list handles and so the application should explicitly release those. This issue will be addressed in a future release.
- When silent Audio is used to create a fingerprint of type gnsdk_fp_type_fapi_nano_query or gnsdk_fp_type_fapi_micro_query, gnsdk_fp_data_get_info() does not return GNSDK_FP_FINGERPRINT_QUALITY_FLAG_SILENT.
- GDO keys such as GNSDK_GDO_CHILD_CONTENT_IMAGEARTIST and GNSDK_GDO_CHILD_CONTENT_BIOGRAPHY that only apply to Contributors but which are also present under Album. They will not return any data when invoked from an Album context.
- Video related functionality in Java and C# modules has not been extensively tested.
- If you are using C++ or Java in your Android development, you must link the libgabi++_shared.so library into your application. This library is available in the Android NDK beginning with version 8d.
- The language codes in gnsdk_manager.h used for setting a locale follow ISO 639-2/B while those returned from a GDO with GNSDK_GDO_VALUE_DISPLAY_LANGUAGE follow ISO 639-2/T. There are a few codes between these two standards that differ.

New in GNSDK 3.07

Gracenote Rhythm

Rhythm is an online Gracenote service that provides track recommendations based on a seed, e.g. artist, album or track. Once a seed has been provided to Rhythm, the application can request additional tracks. The tracks returned may have links to online music catalogs to support playback. The Gracenote Rhythm API supports actions such as “like”, “dislike”, “played” and “skipped” which dynamically influence which tracks are returned on subsequent fetches.

Issues Fixed in 3.07.7

- Fixed Java Security Vulnerability (CVE-2015-2004) in the Java layer of the GNSDK that is potentially exploitable if your application serializes GnException and stores or transmits the serialized object.

Issues Fixed in 3.07.6

- Fixed an issue in SQLite support in which deletion of a large number of items could subsequently lead to errors. Deletion of a large number of items could occur, for example, during background cache management processes.

Issues Fixed in 3.07.5

- Changes applied to GNSDK 3.07.4 in order to allow use of an external SQLite library could lead to database contention. Under some conditions this could cause an application to crash. This has been fixed.
- Fixed a potential crash when performing online queries with multiple threads in parallel.
- Fixed an issue in which client side load balancing was not available when an http proxy was in use.
- Fixed an issue in Playlist generation which impacted the use of LIMIT clauses for tracks in the collection summary in which the attribute to which the LIMIT was applied was missing.
- Fixed a bug in the handling of GNSDK_GDO_VALUE_IDENT for contributor objects.
- Fixed an issue in the C# layer in which an invalid argument exception would be thrown when accessing GnMusicIdFileInfoManager values with non ASCII UTF8 characters.

Changes from GNSDK 3.07.3 to 3.07.4

SQLite for Local Storage

- GNSDK has SQLite built in. This was causing symbol clashes in some cases where developers were using another instance of SQLite in their application. These symbol clashes have been eliminated.
- In addition, a new API has been added that will force GNSDK to use an application specified SQLite instance. The new API is

  gnsdk_storage_sqlite_use_external_library (…)

  This might be desired, for example, to allow sharing databases across processes. Note that in order to use GNSDK with an external version of SQLite, that instance of SQLite may need to be built with the following options

  SQLITE_ENABLE_COLUMN_METADATA

  SQLITE_ENABLE_UNLOCK_NOTIFY

  SQLITE_OMIT_DEPRECATED

Issues Fixed in GNSDK 3.07.4

- Fixed an issue in which Playlist Collection summaries generated with 3.07 version were not compatible with earlier GNSDK 3.06 releases
- Fixed Link Errors for linux_armhf-32 libs
- Fixed an issue in which GNSDK_GDO_VALUE_FULL_RESULT was returning a True value if Classical data was missing even in cases where Classical data was not requested. This key now only returns True for missing Classical data if Classical data was requested and expected.

Issues Fixed in GNSDK 3.07.3

MusicID Stream

- Fixed an issue in which the audio quality setting was sometimes ignored.

Playlist

- Fixed a crash when adding to a pre-existing stored playlist collection which had exactly 3 entries when loaded.

Networking

- Fixed issues with GNSDK_USER_OPTION_NETWORK_TIMEOUT. A recently introduced issue caused this setting to be ignored. In addition, setting it to a very low value, in the low millisecond range, would cause the timeout to be essentially infinite.

Issues Fixed in GNSDK 3.07.2

MusicID, MusicID File

- Rolled back modifications to language preferences for online TOC, Text and file based fingerprint queries. The modifications were causing issues in some regions in which the preferred result was being filtered out. This did not impact streaming fingerprint queries and so the language preference improvements are still in place for those.

Issues Fixed in GNSDK 3.07.1

General

- Removed some diagnostic output which made its way into release builds.

Wrappers

- GnVideo.FindContributors(gnsdk_cstr_t textInput, GnVidoeSearchType searchType)has been deprecated. A new method, taking a search field as well, has been added, GnVideo.FindContributors(gnsdk_cstr_t textInput, GnVideoSearchField searchField, GnVidoeSearchType searchType)

Changes in GNSDK 3.07

Data Model

- The key GNSDK_GDO_VALUE_DISPLAY_SCRIPT_DISPLAY, which was to be used to return the display string of the script of a result, e.g. Latin, has been removed pending provision of supporting data. The key GNSDK_GDO_VALUE_DISPLAY_SCRIPT, which returns the script code, is still available.
- Added the following fields
    - GNSDK_GDO_VALUE_BIRTH_PLACE_LANGUAGE
    - GNSDK_GDO_VALUE_DEATH_PLACE_LANGUAGE
    - GNSDK_GDO_VALUE_CURRENT_MATCH_POSITION_MS

  - GNSDK_GDO_VALUE_MATCH_DURATION_MS
  - GNSDK_GDO_CHILD_TITLE_REGIONAL
  - GNSDK_GDO_CHILD_TITLE_REGIONAL_LOCALE
- Deprecated the following keys
  - GNSDK_GDO_VALUE_TRACK_MATCHED_POSITION. Use GNSDK_GDO_VALUE_MATCH_POSITION_MS instead.
  - GNSDK_GDO_VALUE_ACR_MATCH_POSITION_ACTUAL. Use GNSDK_GDO_VALUE_MATCH_POSITION_MS instead.
  - GNSDK_GDO_VALUE_ACR_MATCH_POSITION. Use GNSDK_GDO_VALUE_MATCH_POSITION_MS instead.
  - GNSDK_GDO_VALUE_ADJUSTED_MATCH_POSITION_MS. Use GNSDK_GDO_VALUE_CURRENT_POSITION_MS instead.
  - GNSDK_GDO_VALUE_ACR_MATCH_POSITION_ADJUSTED. Use GNSDK_GDO_VALUE_CURRENT_POSITION_MS instead.

Locales and Lists

- Logic has been added to prefer query results most consistent with the set locale in cases where there is more than 1 match.
- The Region GNSDK_REGION_NORTH_AMERICA has been deprecated. Use GNSDK_REGION_US instead.

MusicID

- GNSDK_MUSICID_FP_DATA_TYPE_GNFPX has been deprecated. Use GNSDK_MUSICID_FP_DATA_TYPE_STREAM3 or GNSDK_MUSICID_FP_DATA_TYPE_STREAM6 instead.
- GNSDK_MUSICID_FP_DATA_TYPE_CMX has been deprecated. Use GNSDK_MUSICID_FP_DATA_TYPE_FILE instead.

MusicID-Stream

There have been numerous changes to the MusicID-Stream API. Note that much of this API is marked as Experimental and so may change in future releases.

- Renamed some enumeration values from gnsdk_mids_* to gnsdk_musicidstream_* for consistency.
- Added presets to set up MusicID-Stream for different use cases. Currently microphone and radio use cases are supported. This impacts traits such as the assumed audio quality.
  - Added a required “preset” argument to gnsdk_musicidstreamp_channel_create().
- The 5 GNSDK_MUSICIDSTREAM_AUDIO_QUALITY_ option values have been replaced by 2 option values, MICROPHONE and TUNER.
- Added an API to cancel identification, gnsdk_musicidstream_channel_identify_cancel().
- Added an operating mode in which recognition is carried on in the background. This is enabled via a new API, gnsdk_musicidstream_channel_automatic_set() and is primarily

intended to support Radio use cases. In automatic mode, the application can also send events to a MusicID-Stream channel such as broadcast metadata changes.

- Added query flag argument to gnsdk_musicidstream_channel_identify(). Currently only supports GNSDK_MUSICIDSTREAM_QUERY_FLAG_ASYNC.
- APIs added for support of broadcast metadata inputs.

Playlist

- Added a new attribute, GNSDK_PLAYLIST_ATTRIBUTE_NAME_TRACK, to playlist entries.
- Added support for adding empty entries to a Playlist Collection Summaries via a new API gnsdk_playlist_collection_add_ident().

Video

- Added GNSDK_VIDEO_OPTION_ENABLE_MINIMAL_DATA which currently reduces the number of Credits returned from a Video Contributor query.
- Fixed an issue in which the Official name of a contributor could be missing if it was not in the same language as the configured locale.

Platform Support

- Added libraries
  - android_armeabi
  - android_x86
  - iOS_x64
  - linux_armhf_32: 32 bit ARM linux with hard floating point support
- Removed libraries for Winrt and Windows Phone 8. Please contact Gracenote Global Services & Solutions team if you need to develop on these platforms.

Other Improvements

- Made cancellations from query callbacks return Warning instead of Error from all states. Up to now cancellation would sometimes return an error and other times return a warning.
- GNSDK_MANAGER_STORAGE_LISTSCACHE has been deprecated. Please use GNSDK_MANAGER_STORAGE_LISTS instead.
- Added a new value, gnsdk_status_cancelled, to the status callback enumeration type, gnsdk_status_t. This new value indicates to the callback function that a transaction or query has been cancelled.
- Added GNSDK_MANAGER_STORAGE_TEMP to set the location of a new storage for internal use by GNSDK.
- Added iOS and android support for the fp_generation utility application.
- When generating fingerprints for gnsdk_musicidmatch_query_set_compare_fp(), please use the fp utility. It supports all fingerprint types while the gnsdk_dsp library does not.
- Made improvements to load balancing for online queries.

Issues Fixed in GNSDK 3.07

- Fixed an issue in which after calling gnsdk_storage_sqlite_option_set(GNSDK_STORAGE_SQLITE_OPTION_CACHE_FILESIZE, <value>) a new cache was created, the new cache would not respect the file size limit.
- Fixed an in issue in which Android applications using the C API to the GNSDK were required to include the gabi++ library included in the Android NDK. This dependency was not necessary and is no longer required in 3.07. Android applications using the GNSDK C++ and Java APIs will require the gabi++ library from the Android NDK.
- Fixed an issue in which GNSDK_GDO_VALUE_DISPLAY_LANGUAGE_DISPLAY was returning the incorrect field. The language display string returned with this key is now the language corresponding to the code returned by GNSDK_GDO_VALUE_DISPLAY_LANGUAGE.
- Fixed an issue in C++ in which cancel from MusicID was not always effective.
- Fixed an issue in which queries against the cache using GNSDK_LOOKUP_MODE_ONLINE_CACHEONLY would create a cache if none existed.
- Fixed a potential crash when executing *gnsdk_musicidfile_query_do_albumid()* could crash of the flag GNSDK_MUSICIDFILE_QUERY_FLAG_AGGRESSIVE was set.
- Fixed a crashing bug in the query status callback. If a query was aborted when the query state was gnsdk_status_complete, then library shutdown with gnsdk_manager_shutdown() would crash.
- Fixed an issue in which the Official name of a contributor resulting from an online query was missing in some cases where the official language was not the same as the locale language.
- Improved error reporting for querying with silent stream fingerprints. Includes a new error code: GNSDKERR_SilentAudio
- Fixed an issue in with Contributor GNUIDs in MusicID. in which Album and Track IDs were not being properly propagated to Contributor.

Experimental APIs – Do Not Use

Several APIs were added to gnsdk_lookup_local.h in 3.0.7 which are EXPERIMENTAL and should not be used. These are

- gnsdk_lookup_local_storage_create
- gnsdk_lookup_local_storage_open
- gnsdk_lookup_local_storage_release
- gnsdk_lookup_local_storage_add_record
- gnsdk_lookup_local_storage_add_image
- gnsdk_lookup_local_storage_compact2
- gnsdk_lookup_local_storage_set2
- gnsdk_lookup_local_storage_validate2

- gnsdk_lookup_local_storage_get2
- gnsdk_lookup_local_storage_count2
- gnsdk_lookup_local_storage_get2
- GNSDK_LOOKUP_LOCAL_STORAGE_PRIMARY_IDENTIFIER

Changes from GNSDK 3.06

User Registration

User registration is now explicit rather than being a hidden step in the first query. An application can specify whether the user should be registered for online and local queries, or just for local queries. Registration for online queries requires a request to Gracenote Service. In addition, a user handle can be queried directly to see if it is enabled for online queries. All of this allows the application to be certain that when it attempts to do an online query, it has a user that is registered to do so, thereby avoiding query errors. It also gives the application control of when online registration is performed.

Note

An application should register online with Gracenote only once per user.

You must create a user handle to perform queries against local or online Gracenote databases. Each device must register itself with Gracenote one time only. Use the returned registration data to create the user handle.

The previous user registration functions:

- *gnsdk_manager_user_create()*
- *gnsdk_manager_user_create_new()*

are replaced by these functions:

- *gnsdk_manager_user_register(): (New)* Registers a new user and returns the serialized user data. Call once for the life of a device and store the serialized data. A user can be registered for local lookups only or for online and local lookups. Registration for local and online requires an online connection. If registered for local only, you can register for local and online at a later time and replace the local online serialized user data without any detriment.
- *gnsdk_manager_user_set_autoregister(): (New)* Sets a callback function that will be called when a user handle needs to be stored locally.
- *gnsdk_manager_user_create(): (Changed). Creates a user handle from a previously registered, serialized user.*
- *gnsdk_manager_user_is_local_only(): (New)* Indicates whether whether a user handle is only valid for local lookups, or is valid for online lookups as well.

These functions are supported by these new constants:

- *GNSDK_USER_REGISTER_MODE_ONLINE*
- *GNSDK_USER_REGISTER_MODE_LOCALONLY*

In addition, the *previously* deprecated functions for user serialization have been removed:

- *gnsdk_manager_user_serialize()*
- *gnsdk_manager_user_serilaize_needed()*
- *gnsdk_manager_user_release()* - no longer returns a serialized user

Gracenote recommends that applications use the autoregister callback mechanism to register.

Online Processing for AlbumID

The new online MusicID-File query option, GNSDK_MUSICIDFILE_OPTION_ONLINE_PROCESSING, enables AlbumID to group tracks into albums using Gracenote Service instead of on the client. This option greatly improves AlbumID performance by reducing the amount of client-side processing and number of queries to Gracenote Service.

Two new match types support this feature: MIDF_MATCH_TYPE_ONLINE_ALBUMID_FP and MIDF_MATCH_TYPE_ONLINE_ALBUMID_TEXT. These types indicate which query type was the most authoritative matching type for a given file. For more information, please see the API Reference description for the GNSDK_GDO_VALUE_TRACK_MATCH_TYPE value key.

Locales and Lists

The following enhancements have been made to the locale and list system:

- The ability to set the locale per query has been added through the new API
  - *gnsdk_musicid_query_set_locale()*
- Added the status callback support for list and locale updates through the new API
  - *gnsdk_manager_list_update_notify()*
  - *gnsdk_manager_locale_update_notify()*
- The following GNSDK_LANG_* constants have been renamed.
  - GNSDK_LANG_PORTUGUESE_BRAZIL changed to GNSDK_LANG_PORTUGUESE
  - GNSDK_LANG_BAHASA_INDONESIA changed to GNSDK_LANG_INDONESIAN
- The following region definitions have been added
  - GNSDK_REGION_NORTH_AMERICA
  - GNSDK_REGION_LATIN_AMERICA
  - GNSDK_REGION_INDIA

MusicID-Stream

MusicID-Stream is a new module designed specifically to recognize music delivered as a continuous stream. Use this library in applications that need to recognize streaming music in real time and on-demand. MusicID-Stream receives and efficiently manages incoming audio making it ready for fast identification. To learn how to use this module, refer to the documentation and the musicid_stream sample application.

Java, C# and C++ APIs

The Java, C# and C++ APIs for GNSDK 3.05 was released as Beta. After overwhelming feedback from all of you, we have done our best to incorporate them in GNSDK 3.06.

Other Improvements

- **GNSDK_FP Utility:**
  - Modified to a more flexible callback model for returning raw fingerprint data
  - Added support for streaming ('fapi') fingerprint types
- **MusicID** : The text input GNSDK_MUSICID_FIELD_ARTIST has been deprecated. Use the alternative fields GNSDK_MUSICID_FIELD_TRACK_ARTIST and GNSDK_MUSICID_FIELD_ALBUM_ARTIST instead.
- **GDO Type Checking:** Added an API to determine the type of a GDO, *gnsdk_manager_gdo_is_type()* which returns an SDKMGRERR_HandleObjectWrongType error if gdo_handle is not of the specified type.
- **Online Queries:** Added options GNSDK_USER_OPTION_NETWORK_INTERFACE to gnsdk_manager.h. For devices with multiple Network Interfaces, this allows setting which NIC is used by Gracenote online features during a given session. The NIC is selected through its IP address. In order to select the NIC on a per query basis, use the option "gnsdk_queryopt_network_intf".
  - IPv6 addresses have not been tested.

- APIs that will be deprecated in future are now clearly marked using a platform specific macro, GNSDK_DEPRECATED_API.
- Added a new API, gnsdk_manager_test_gracenote_connection(), to test the ability to connect to Gracenote Service.
- Removed the following keys related to alternate name retrieval. Feature support has been moved to Nuance.
  - GNSDK_MUSICID_OPTION_ENABLE_ALTERNATE_NAMES
  - GNSDK_MUSICIDFILE_OPTION_ENABLE_ALTERNATE_NAMES
  - GNSDK_GDO_CHILD_NAME_ALTERNATE
  - GNSDK_GDO_CHILD_TITLE_ALTERNATE
- The gnsdk_manager_gdo_render_to_xml() function has been renamed to gnsdk_manager_gdo_render(). It now supports JSON rendering, in addition to XML.
- Added a new API, gnsdk_manager_gdo_create_from_xml(), to reconstitute a GDO from rendered XML.
- The match position key for MID-Stream, GNSDK_GDO_VALUE_TRACK_MATCHED_POSITION, has been renamed to GNSDK_GDO_VALUE_MATCH_POSITION_MS.

- Track search using gnsdk_musicid_query_find_tracks() will not be supported in the future and has been marked for deprecation.
- The android_arm-32 folder has been renamed to android-armeabi-v7a.

Issues Fixed in GNSDK 3.06

- Fixed MoodGrid coordinates by swapping X and Y so as to match the intended design.
- Fixed an issue in which a crash could occur when shutting down MoodGrid using gnsdk_moodgrid_shutdown().
- Fixed a possible segmentation fault at GNSDK shutdown. The segmentation fault could occur if certain resources were still held by the application when shutdown was called.
- Fixed a crashing bug in the query status callback. If a query was aborted when the query state was gnsdk_status_complete, then library shutdown with gnsdk_manager_shutdown() would crash.
- Fixed a deadlock that could occur in some circumstances when attempting to update an expired online cache item.
- Fixed a potential crash when executing *gnsdk_musicidfile_query_do_albumid()* could crash of the flag GNSDK_MUISICFILE_QUERY_FLAG_AGGRESSIVE was set.
- Fixed a bug in which with gnsdk_link_query_option_set(GNSDK_LOOKUP_MODE_ONLINE), if the image to be fetched was expired in the cache, the SDK would not go online. This resulted in no image being returned. The correct behavior is to fallback to an online query to refresh the expired cache record.
- Fixed an issue in which queries against the cache using GNSDK_LOOKUP_MODE_ONLINE_CACHEONLY would create a cache if none existed.
- Fixed an issue with gnsdk_storage_release() not closing the associated file.
- Fixed an issue in which calling the gnsdk_manager_storage_flush() function could later cause SQLite shutdown to fail due to unreleased handles.
- Fixed the following issues related to setting GNSDK_MUSICID_OPTION_LOOKUP_MODE to GNSDK_LOOKUP_MODE_ONLINE_CACHEONLY:
    - Setting this through gnsdk_musicid_query_option_set() was returning Invalid Argument error.
    - Setting this through gnsdk_musicid_user_option_set() would cause unsupported functionality to be returned by a subsequent call to gnsdk_link_query_create().
    - Even with ONLINE_CACHEONLY mode set, if no item was found in the cache, GNSDK would automatically perform an online query. In this mode, GNSDK should not go online.
- Fixed a bug that incorrectly returned an error after a failed attempt to write an image to the content cache due to the cache's size limit. This action should not return an error because the query itself succeeded and all data is still available.

- Fixed an issue in which gnsdk_manager_gdo_value_count() could return the wrong value.
- Fixed a bug in which genre art could not be retrieved for an album response returned from an online lookup.

Confidentiality Notice

This document is confidential information of Gracenote, Inc., and is for Gracenote employees and intended recipients only. Any dissemination, distribution, or copying of this communication is strictly prohibited.