# NGSQuickSynth Library Overview



PlayStation®Vita Programmer Tool Runtime Library Release 3.50

# Table of Contents

**1 Library Overview ..... 3**

Characteristics ..... 3

Files ..... 3

Kernel Resources used by NGSQuickSynth ..... 4

Integrating NGSQuickSynth Library ..... 4

Sample Code ..... 4

Related Documentation ..... 4

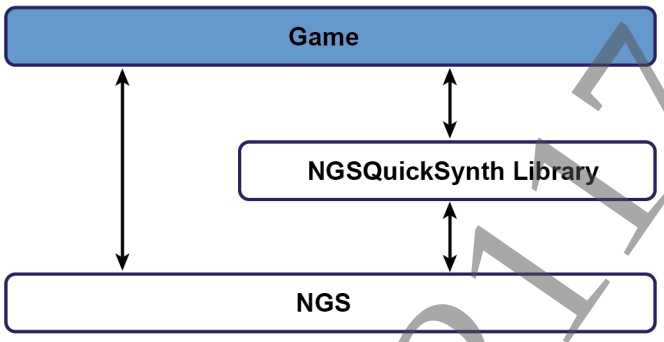
**2 Using the Library ..... 5**

# 1 Library Overview

## Characteristics

The NGSQuickSynth Library is an audio configuration system designed to provide application developers with a simple way to set up and update the audio synth, while still allowing them to access NGS directly to control the operation.

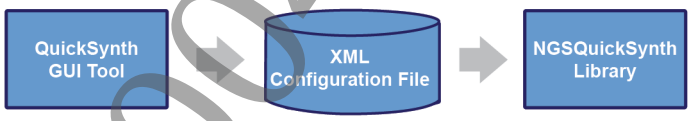
**Figure 1 NGSQuickSynth Library Interaction**




The library reads the audio system configuration information from an XML file, which can easily be generated using the QuickSynth PC Tool, included in the SDK.

QuickSynth is a tool that provides a graphical user interface to configure the NGS synth for PlayStation®Vita. It allows you to configure an audio system easily without a deep understanding of the library interface, while allowing you to customize the library as much as you need from a simple and intuitive user interface. For complete details, refer to the *QuickSynth Tool User's Guide*.

**Figure 2 QuickSynth System Overview**




You can create a customized Audio System using QuickSynth, export the configuration to a XML file and use it to configure NGS at runtime.

## Files

The files required to use the QuickSynth API are listed in Table 1.

**Table 1 QuickSynth Files**

| File Name | Description |
|---|---|
| `quicksynth.h` | Main NGSQuickSynth library header file. |
| `quicksynth\error_codes.h` | Error return codes header file. |
| `quicksynth\custom_memory.h` | Memory management function definitions header file. |
| `libSceNgsQuickSynth.a` | Static library file. |

Note that QuickSynth requires the use of other modules in the system:

- NGS: Audio Synthesizer.
- Sulpha: NGS debugger (only if NGS debugging information is required).

## Kernel Resources used by NGSQuickSynth

The NGSQuickSynth Library creates a lightweight mutex which is used to ensure thread safety. This mutex is deleted when the QuickSynth object is destroyed.

## Integrating NGSQuickSynth Library

To integrate the library:

(1) Include "`quicksynth.h`". Various other headers will be included automatically.

(2) Load the PRK modules in the program – NGS, Sulpha (if required for debugging). For example:

```
int returnCode = sceSysmoduleLoadModule(SCE_SYSMODULE_NGS);
if (returnCode != SCE_OK) {
    return returnCode;
}
```

(3) Link `libSceNgsQuickSynth.a` and `libSceNgs_stub_weak.a` into the program.

## Sample Code

Sample code that demonstrates the library API and some sample configuration files are available in: `sample_code\audio_video\api_libquicksynth`.

## Related Documentation

Refer to the following related documents, included in the SDK, for further information on the following areas:

- NGSQuickSynth Reference: See the *NGSQuickSynth Library Reference*.
- QuickSynth Tool: See the *QuickSynth Tool User's Guide*.
- NGS system: See the *NGS Overview* and *NGS Reference*.
- NGS modules: See the *NGS Modules Overview* and *NGS Modules Reference*.
- Sulpha: See the *libsulpha Overview* and *libsulpha Reference*.

# 2 Using the Library

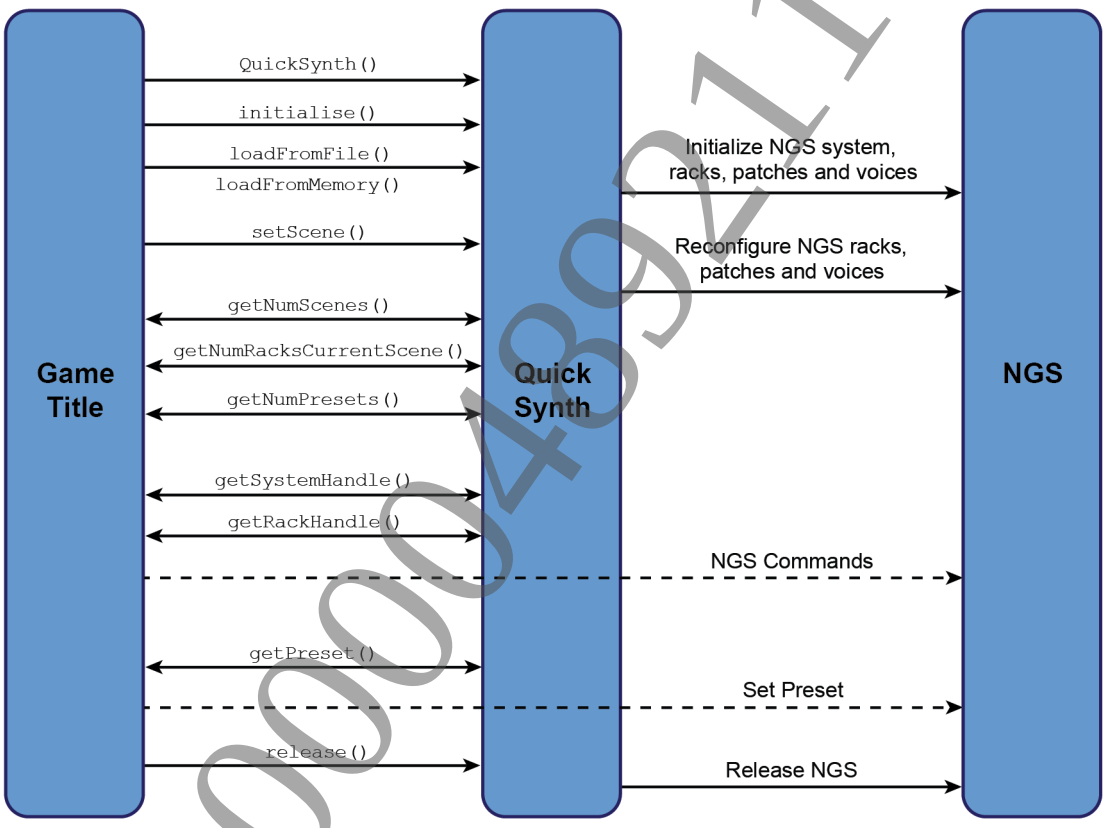
The NGSQuickSynth library provides a simple interface to set up and control NGS. To initialize the system, you just need to create an instance of QuickSynth, call the initialization function and load the configuration file.

The system is initially configured to use the "default" scene but a different one can be set at any time, by calling `setScene()`.

The next step is to start the Audio Update Thread and initialize the Audio Output module. At this stage, the system is ready to run and it can be controlled by getting handles to the NGS system, racks and voices and using the NGS commands directly (for example, `sceNgsVoicePlay`).

The presets in the QuickSynth configuration file can be retrieved and set to any voice dynamically at any time, using the NGS interface directly.

**Figure 3 NGSQuickSynth Library Operation**




See the NGSQuickSynth Library Sample code in the SDK for an example of the library operation.