

# Quick Start

## Setting Up TWL-System

2009/03/04

**The content of this document is highly confidential  
and should be handled accordingly.**

**Confidential**

These coded instructions, statements, and computer programs contain proprietary information of Nintendo of America Inc. and/or Nintendo Company Ltd. and are protected by Federal copyright law. They may not be disclosed to third parties or copied or duplicated in any form, in whole or in part, without the prior written consent of Nintendo.

## Table of Contents

---

1	Introduction .....	5
2	TWL-System Setup .....	6
2.1	Expanding the TWL-System Package .....	6
2.2	Configuring the Environment Variables .....	6
2.3	Setting Up a Run Environment for TWL Tools .....	6
3	Building the Libraries and Sample Programs .....	7
3.1	Development Tools Needed.....	7
3.1.1	The Build Environment.....	7
3.2	Building the TwlSystem Tree.....	7
4	Running Sample Programs.....	9
4.1	Running on ensata.....	9
4.1.1	Starting Ensata .....	9
4.1.2	Loading and Running Programs .....	9
4.2	Running with IS-NITRO-DEBUGGER .....	9
4.2.1	Starting IS-NITRO-DEBUGGER .....	9
4.2.2	Loading and Running Programs .....	9
4.3	Running with IS-TWL-DEBUGGER .....	10
4.3.1	Starting IS-TWL-DEBUGGER .....	10
4.3.2	Loading and Running Programs .....	10
4.4	Debugging with CodeWarrior.....	10
4.4.1	Loading and Running Programs .....	10

## Revision History

Revision Date	Description
2009/03/04	Revised content related to starting ensata. Revised content related to the build environment. Corrected a typo.
2008/05/30	Revisions resulting from NITRO-System name change (updating to TWL-System).
2008/04/11	Revised content for section 2.2 Configuring the Environment Variables
2008/04/08	Initial version. (Separated from <code>BuildSystem.pdf</code> .)

# 1 Introduction

This document describes the setup process for TWL-System as well as the procedure for building its libraries and demo programs.

## 2 TWL-System Setup

### 2.1 Expanding the TWL-System Package

---

Expand the TWL-System package anywhere on a local drive. The TWL-System package is compressed in the ZIP format, so a wide variety of expansion utilities should be able to open it. Once the package has been expanded, there is a directory on that drive named `TwlSystem`.

### 2.2 Configuring the Environment Variables

---

The tools and library build system that come with TWL-System get the location of TWL-System on a drive from the environment variable. Configure the absolute path to the `TwlSystem` directory as the environment variable `TWLSYSTEM_ROOT`. Hereafter, this document refers to this directory as `$TwlSystem`.

### 2.3 Setting Up a Run Environment for TWL Tools

---

The applications available with TWL-System have been tested and confirmed to run on the Microsoft Windows XP Service Pack 2 operating system. Testing has not yet been performed on the Microsoft Windows Vista operating system so we cannot guarantee that our applications will run properly.

The use of NITRO-SoundMaker requires Microsoft .NET Framework 1.1. Because the TWL-System package does not include .NET Framework, you need to install it if it has not yet been installed on the computer you are working with. Note that we can only guarantee confirmed results with .NET Framework 1.1.

Any attempts to launch NITRO-SoundMaker without having installed .NET Framework 1.1 results in application errors, and an inability to run the software. Should this happen, confirm that .NET Framework 1.1 has been properly installed on your computer. To install it, use the **Add or Remove Programs** tool in the Windows XP Control Panel. Check to see that .NET Framework 1.1 is included in the **Currently installed programs** list.

## 3 Building the Libraries and Sample Programs

### 3.1 Development Tools Needed ---

The TWL-System library build environment is built on top of the TWL-SDK's build environment. As such, the build environment for TWL-System libraries is identical to one for the SDK. To use the TWL-System, you need to have a TWL-SDK environment in place.

#### 3.1.1 The Build Environment ---

TWL-System builds have been tested and confirmed to work on the Microsoft Windows XP Service Pack 2 operating system. The following tools and SDK are needed to build and debug the TWL-System libraries and any applications that use those libraries.

- CodeWarrior for Nintendo DS
- Cygwin
- TWL SDK
- NITRO emulator (ensata)
- IS-NITRO-EMULATOR (to run NITRO platform binaries)
- IS-TWL-EMULATOR (to run TWL platform binaries)

### 3.2 Building the TwlSystem Tree ---

Once you have all the necessary development tools, launch Cygwin. Libraries and sample programs can both be built by running make commands in the TWL-System library's root directory (the `TwlSystem` directory), using the Cygwin (bash) shell.

The `TWLSDK_PLATFORM` environment variable needs to be configured for your target platform when building. Enter the following to build the NITRO platform's release version of its libraries and sample programs.

```
cd $TwlSystem
make TWLSDK_PLATFORM=NITRO
```

To build libraries and sample programs other than release builds, you need to explicitly configure the compile target. Enter the following to build the debug build version of the libraries and sample programs for the TWL platform.

```
cd $TwlSystem
make TWLSDK_PLATFORM=TWL TWL_DEBUG=TRUE
```

Run a `make` command in the `$TwlSystem/build/demos` directory to build only the sample programs. Enter the following to build the debug build version of the sample programs only for the TWL platform.

```
cd $TwlSystem/build/demos
make TWLSDK_PLATFORM=TWL TWL_DEBUG=TRUE
```

For more information on the environment variables requiring configuration when building, as well as the specifiable build switches, see the following documentation in the TWL-SDK.

`$TWLSDK_ROOT/docs/SDKRules/ Rule-Defines.html`



## 4 Running Sample Programs

Test sample programs to confirm that their build occurred normally.

### 4.1 Running on ensata ---

Use the NITRO emulator ensata to run NITRO builds in TWL-System. The following describes the procedures involved in using ensata.

#### 4.1.1 Starting Ensata ---

To start ensata, double-click the ensata icon. To use ensata, either DirectX8 or later (when using DirectInput and DirectSound only) or DirectX9 or later (when using Direct3D features) is required. If your computer does not have these installed, they are available for download at Microsoft's website.

#### 4.1.2 Loading and Running Programs ---

1. From the **Run** menu, select **Load a NITRO file**. The **Open File** dialog box appears.
2. From the **File Type** list, select **Binary Files (\*.bin, \*.srl)** and, in the **File Name** field, enter the SRL file you want to run. Click **Open** to load that file in ensata.
3. To run a loaded program in ensata, click the **Run** button (in the upper left) in the ensata window. Use the second button from the left, **Stop**, to halt a running program as well.

### 4.2 Running with IS-NITRO-DEBUGGER ---

Use IS-NITRO-DEBUGGER, the NITRO debugger, to run and debug NITRO builds in the TWL-System. The following describes a simple procedure for using IS-NITRO-DEBUGGER to run a program. For more information on the use of IS-NITRO-DEBUGGER, see its documentation.

#### 4.2.1 Starting IS-NITRO-DEBUGGER ---

Begin by turning the IS-NITRO-EMULATOR on and opening the IS-NITRO-EMULATOR controller. Next, double-click the IS-NITRO-DEBUGGER (ISND.exe) icon to launch the debugger software for IS-NITRO-EMULATOR.

#### 4.2.2 Loading and Running Programs ---

1. From the **File** menu, select **Open**. The **Open File** dialog box appears.
2. Select SRL as the type of file to open and, in the **File Name** field, enter the SRL file you want to run. Click **OK** to load that file in the debugger and reset the hardware.

To run a program loaded to the debugger, press F9. Press ESC to stop a program that is currently running.

---

## 4.3 Running with IS-TWL-DEBUGGER

---

Use IS-TWL-DEBUGGER, the TWL debugger, to run and debug TWL builds in the TWL-System. The following describes a simple procedures involved in using IS-TWL-DEBUGGER to run a program. For more information on the use of IS-TWL-DEBUGGER, see its documentation.

---

### 4.3.1 Starting IS-TWL-DEBUGGER

---

Begin by turning the IS-TWL-EMULATOR on and opening the IS-NITRO-EMULATOR controller.

Next, double-click the IS-TWL-DEBUGGER (`ISND.exe`) icon to launch the debugger software for IS-TWL-EMULATOR.

---

### 4.3.2 Loading and Running Programs

---

1. From the **File** menu, select **Open**. The **Open File** dialog box appears.
2. Select SRL as the type of file to open and, in the **File Name** field, enter the SRL file you want to run. Click **OK** to load that file in the debugger and reset the hardware.

To run a program loaded to the debugger, press F9. Press ESC to stop a program that is currently running.

---

## 4.4 Debugging with CodeWarrior

---

A debugger for the DS is included in the IDE for CodeWarrior for Nintendo DS. The CodeWarrior debugger can be used to debug programs running on either IS-NITRO-DEBUGGER or ensata. The following describes a simple procedures involved in using CodeWarrior to run a program. For more information on the use of the CodeWarrior IDE, see its documentation.

---

### 4.4.1 Loading and Running Programs

---

1. To launch the IDE, double-click the CodeWarrior IDE (`IDE.exe`) icon.
2. From the **File** menu, select **Open**. The **Open File** dialog box appears.
3. Enter the name of the NEF file you want to run in the **Object name** field. Click **Open**, and a **Choose Debugger** dialog box appears.
4. Select the debugger you want to use and then click **OK**. A project (MCP) file is automatically created, and the program loads.
5. To start the IDE debugger, press F5. If you then click the **Run** button, at the furthest left, the program runs in the debugger. Click the **Break** button, second from the left, to halt a running program.

Windows is a registered trademark or trademark of Microsoft Corporation (USA) in the United States and elsewhere.

All other company and product names are the trademarks or registered trademarks of their respective companies.

© 2004-2009 Nintendo

The contents of this document cannot be duplicated, copied, reprinted, transferred, distributed, or loaned in whole or in part without the prior approval of Nintendo.