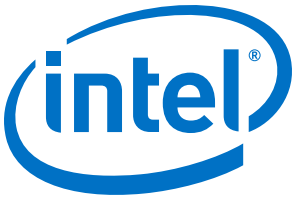


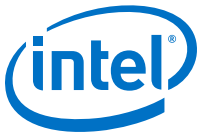
# Intel® IoT Gateway Diagnostics Tool

## User Guide

*October 2015*

**Intel Confidential**

Doc. No.: 557818, Rev.: 1.0



# Revision History

| Date | Revision | Description |
|---|---|---|
| October 2015 | 1.0 | Initial public release |

# Contents

# 1.0 Introduction

## Applicable to These Intel® IoT Gateway Products

- Intel® IoT Gateways based on Intel® Atom™ processors
- Intel® IoT Gateways based on Intel® Core™ processors

## How These Instructions Help You

This document helps you set up a test environment to run diagnostic tests using a Graphical User interface (GUI). This lets you to check the presence, configuration, and basic functionality of various gateway components, using both automated and manual tests. For test descriptions, see Diagnostics Tool Test Descriptions.

## Document Terminology and Conventions

- **Terminology**
  - Gateway: Hardware included in your gateway kit.
  - Intelligent Device Platform: Bundled software from Wind River Systems, Inc.
- **Conventions**
  - `This font is used for commands, API names, parameters, filenames, directory paths, and executables.`
  - **Bold text** is used for graphical user interface entries, buttons, and keyboard keys.

```
This font in a gray box is used for commands you must type or include in a script.
```

## Items You Need to Provide

- An installed Intel® IoT Gateway.
  - For hardware installation steps, see https://software.intel.com/en-us/SetupGateway-hardware.
  - To install the Wind River tools and set up your Development Computer, see https://software.intel.com/en-us/Setup-IDP-DevelopmentTools.
- A computer running a Windows operating system to use as the UI Development Computer. This computer must have Internet access and a current Internet browser installed on it.
- One 4 GB or larger USB flash drive formatted with the FAT32 file system.

## Reporting Technical Issues

For any technical issues with the Diagnostics Tool, open an Intel® Premier Support (IPS) ticket under product code **Intel IoT Gateway Products**.

# 2.0 Install the Diagnostics Tool on the Gateway

1. Confirm you received an email message containing a copy of your license agreement and review the agreement.
2. Power on your gateway and login using `root` as both the user ID and the password.
3. If you are behind your corporate firewall, you may need to set up a proxy. Contact your IT department for your proxy IP address and port. Then set it up on your gateway, replacing `proxy server IP address:port` with the IP address and port for your proxy:

```
export http_proxy=http://proxy server IP address:port
```

4. Install the Diagnostics Tool on your gateway:

```
smart channel --add diag type=rpm-md name=diag_REPO baseurl=http://
198.175.66.251/WRL/7/extras
smart update diag
smart install diag
```

*Note:* During the installation you might see the message `/sbin/ldconfig: /lib64/libubus.so.0 is not a symbolic link`. Disregard this message; it does not indicate an error.

The Diagnostics Tool is installed on your gateway's hard drive in `/opt/Intel/Diagnostics`. It integrates with the UI that is installed on a separate computer.

The test folders under `/opt/Intel/Diagnostics/HostScripts/WRL7-scripts` each contains two files:

- `Attributes.xml`: Test attributes required for UI integration.
- `README.TXT`: Basic test information, test procedures, and test categories.

The automated test folders contain the following files at a minimum:

- `setup.sh`: Script that includes common code and ties the test to the Web GUI (json objects). This is a test setup and initialization script that calls `qualify.sh`
- `qualify.sh`: Script that executes a test procedure and reports the results to `setup.sh`.

# 3.0 Install the Diagnostics Tool on the UI Computer

1. Copy the gateway directory `/opt/Intel/Diagnostics/UI-Windows-Installer` from the gateway to the USB flash drive:

   *Note:* When the USB flash drive is inserted into the gateway, the FAT32 partition will be auto-mounted at mount point `/media/sdX1`, where `X` represents the USB flash drive letter.

2. Copy `UI-Windows-Installer` from the USB flash drive to a directory on the UI Computer.
3. On the UI Computer, use administrator privileges to open a command prompt window.

   *Important:* You will not be able to complete the installation if you do not access the command prompt window with administrator privileges.

4. Change to the `UI-Windows-Installer` directory on the UI Computer.
5. While still at the command prompt, type `InstallHealthCheck.bat` and press **Enter**.
6. If a **User Account Control** window displays, select **Yes**.
7. When prompted to do so, press a key to continue. The **Mono for Windows Setup** window opens.

   *Important:* Select **No** if an alert dialog displays with a message similar to `Do you want to allow the following program to make changes to this computer?`. This message indicates you did not open the command window with administrator privileges. Return to step 3.

8. Review the license agreement. If you agree to the terms, click the acceptance box and then click **Install**.
9. When the installation completes, click **Finish**.
10. A caution screen displays regarding registry changes. Click **Yes**.
11. A message screen displays to inform you that the registry was changed. Click **OK**. You are returned to the command prompt screen.
12. In the existing command prompt window, navigate again to the `UI-Windows-Installer` directory.
13. Type `Intel.Gateway.Health.Check.UI.Setup.msi`. Press **Enter** and then click **Next** to continue.

### Install .NET Framework Version 4.5

*Note:* The screen and text you see may vary slightly, depending on your Internet browser.

1. Click download. The message **Thank you for downloading** displays and the download begins.

*Note:* If the download does not begin, click **Click here** in the Internet browser to begin the download.

2. Your screens and prompts will vary, depending on your Internet browser. Follow the on-screen prompts to run the installation software. Upon completion, the **.NET FrameWork 4.5 Setup** screen displays.
3. Review the license agreement. If you agree to the terms of the agreement, click the acceptance box and then click **Install**.
4. If prompted to close your Internet browser, click **Yes**.
5. On the **Installation is Complete** screen, click **Finish**.
6. Restart your UI Computer.
7. Use administrator privileges to open a command prompt window.
8. Change to the directory into which you copied the contents of `/opt/intel/diagnostics/UI-Windows-Installer`.
9. Type `Intel.Gateway.Health.Check.UI.Setup.msi` and press **Enter**. The Diagnostics User Interface loads.
10. Continue to Install the Diagnostics User Interface.

### Install the Diagnostics User Interface

1. On the welcome screen, click **Next**.
2. On the **Select Installation Folder** screen, click **Next**.
3. On the **Confirm Installation** screen, click **Next**.
4. On the **Installation Complete** screen, click **Close**.

# 4.0 Running Diagnostics Tests

## Start Diagnostics Console Program on Gateways

Perform these steps on each gateway to be tested.

1. Login to the gateway using `root` as both the username and password.
2. Start the diagnostics agent:

```
diag
```

This response indicates the gateway is ready:

```
TCP listening address = 0.0.0.0 on port 5150.
UDP listening address = 0.0.0.0 on port 5150.
Starting HostService service in console mode.
Press 'q' to quit…
```

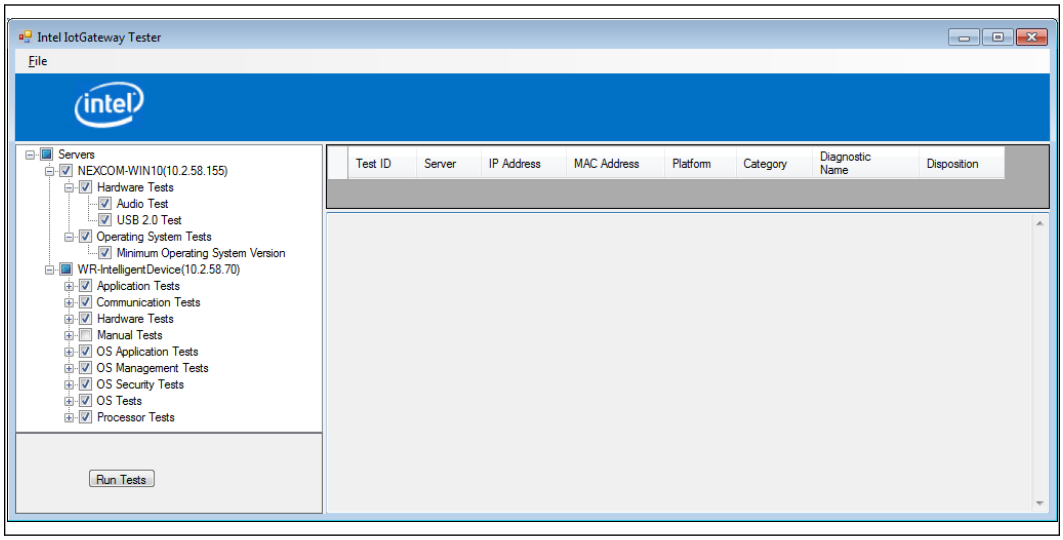
## Run Automated Tests

*Important:* Start all gateways to be tested before beginning this section. Otherwise, the gateways will not be auto-discovered.

1. On your UI Computer, go to `C:\Program Files (x86)\Intel\Intel Gateway HealthCheck UI`.
2. Run `Intel.Gateway.Health.Check.UI.exe`

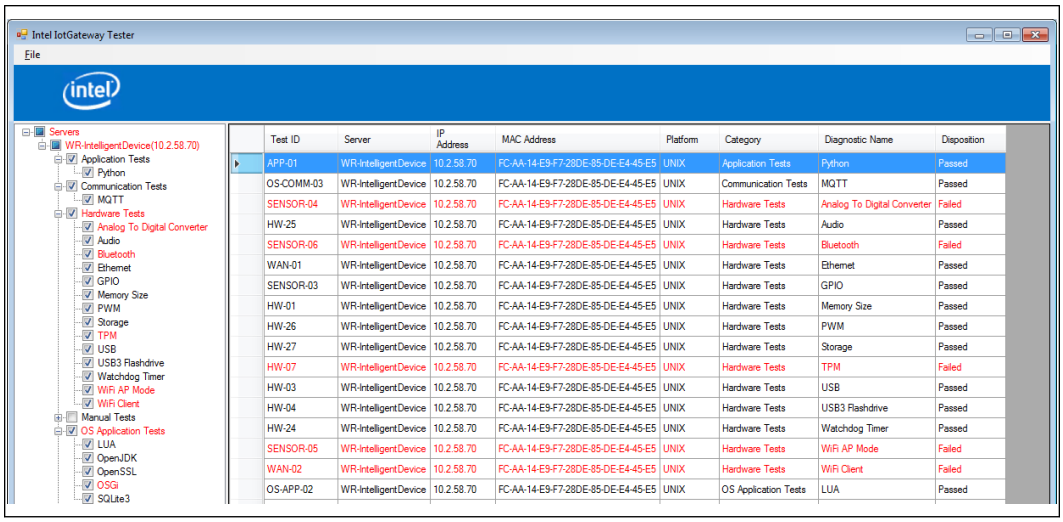
   The **Intel IotGateway Tester** opens and displays the servers (gateways) that are on the same subnet as the UI Computer. Under each gateway is a list of tests that you can select.

   *Note:* If you do not see a gateway listed, either that gateway is on a different subnet, or you did not complete Start Diagnostics Console Program on Gateways. If you completed the steps above and do not see a connected gateway in the list, see Manually Discovering Gateways.




3. Expand and collapse the categories using the **+** and **-** symbols and click the checkbox for each test that you want to run.
4. Click **Run Tests.**

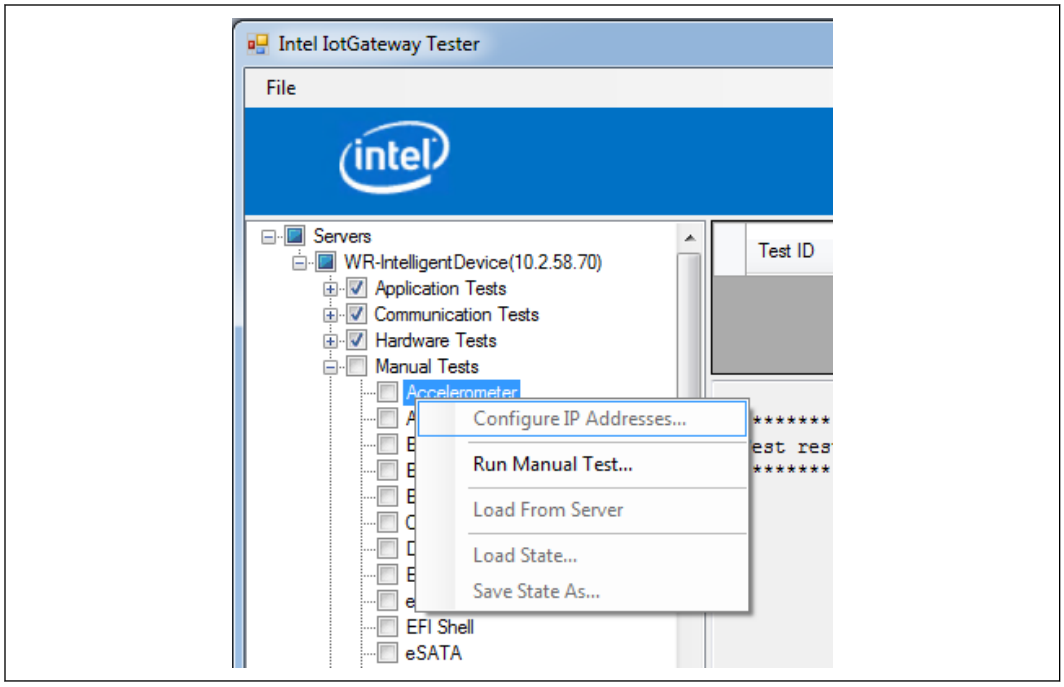
   Your results will appear as follows, with failed tests appearing in red:




## Running Manual Tests

Manual tests are provided for tests that require human intervention and interpretation.

1. To run a manual test, use the **+** next to **Manual Tests** to view the list of tests.
2. Right-click the test you want to run and select **Run Manual Test...** from the menu that opens.




3. Each manual test has an associated test procedure. To complete the test, follow the directions in the test procedure, enter **Comments**, and select **Pass**, **Fail**, or **Blocked**. Then click **Save**.

   *Note:* The **Blocked** option is enabled only after the blocked condition is described in the **Comments** box.

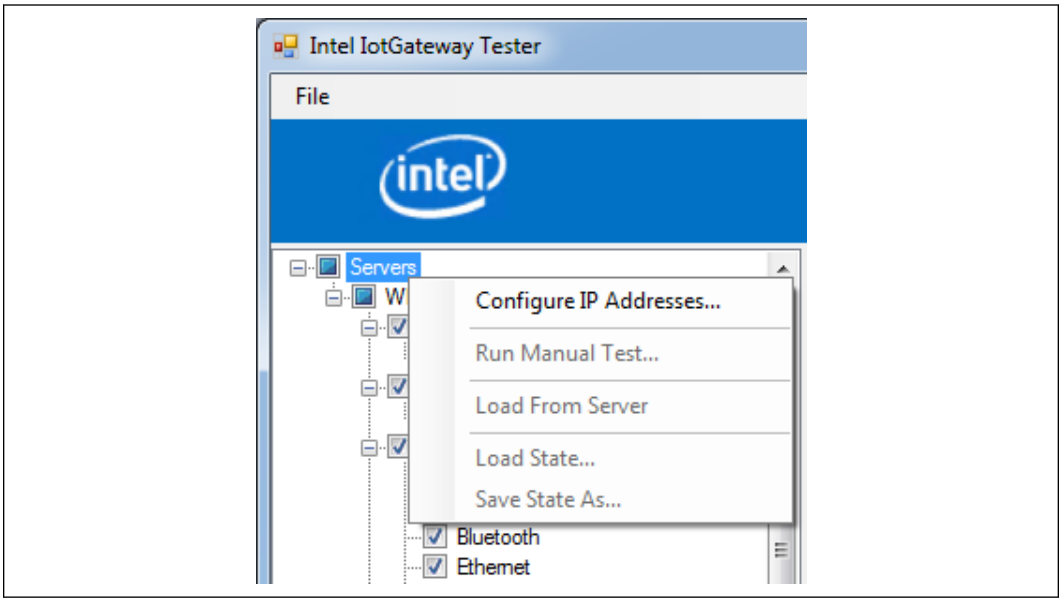
# Appendix A Manually Discovering Gateways

Gateways on the same subnet that are not auto-discovered and gateways that are on different subnets can be manually included by explicitly specifying the gateway IP address.

1. On the gateway, determine the IP address for that gateway:

```
ifconfig
```

2. Write down the IP address that applies to `inet addr` for `eth0`.
3. Return to the UI Computer.
4. Right-click on **Servers**.
5. Select **Configure IP Addresses...**



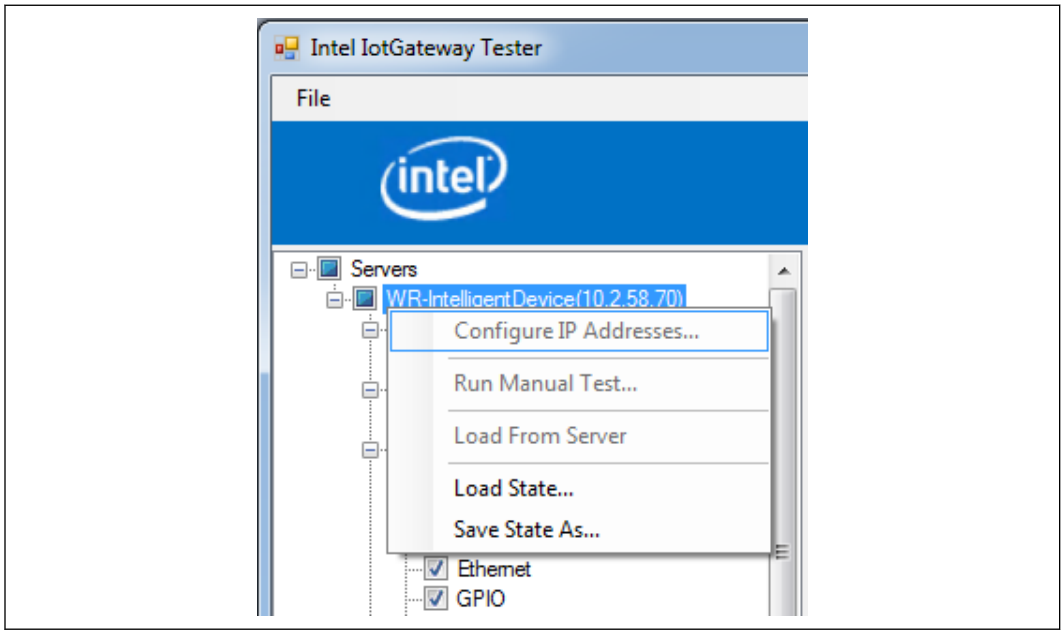

6. Enter the **IP Address** of the gateway and an identifier in the **Server Name** field.

# Appendix B Save and Load State

You can save and load the state of the checkmarks for your gateways. This enables you to quickly load and repeat a series of tests.

## Save State

1. Right-click the gateway for which you want to save the state of the checkmarks (checked or unchecked).
2. Click **Save State As**.




3. Browse to the desired location and enter a file name.

## Load State

A previously saved state can be loaded and applied to *any* gateway.

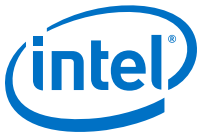
1. Right-click the gateway for which you want to load the previously saved state.
2. Browse to and select the file.

# Appendix C Diagnostics Tool Test Descriptions

The tests below are listed in the `Diagnostics` directory under `/opt/Intel/Diagnostics/<test_name>` and the test procedures are in the `README.TXT` file in each directory. You must test each component in the table that is labeled as **Required** in the **Test Type** column.

| Test Name | Test ID | Test Type | Description |
|---|---|---|---|
| Accelerometer | SENSOR-11 | Optional Manual | Check for the presence of the driver for the accelerometer.<br>1. Confirm the presence of a customer-supplied driver.<br>2. If present, the driver should contain a sensor from which the values can be read.<br>3. Pass if the driver is present and the values can be read. Otherwise FAIL. |
| Analog_To_Digital_Converters | SENSOR-04 | **Required** Automated | If this test confirms are any ADC devices are present, then the IIO bus will become available. Compare the measurement data to the acceptable range. |
| Antenna Mounting | HW-11 | **Required** Manual | Look for a provision on the board for antenna mounting. |
| Audio | HW-25 | **Required** Automated | Confirm the presence of audio devices on the gateway. This test requires an audio clip. The test performs the following steps:<br>1. Detect audio devices. FAIL if none found.<br>2. Display detailed information about found audio hardware.<br>3. Provide device driver information.<br>4. Play audio sample.<br>5. Display codec information. |
| BIOS Password | FIRM-01 | Recommended Manual | Check that the system firmware supports the ability to password-protect entry into the firmware setup menu. |
| BIOS UEFI Architecture | FIRM-06 | **Required** Manual | Check the BIOS/UEFI architecture.<br>1. Check with the BIOS vendor to confirm the architecture is 64-bit.<br>2. PASS if the architecture is 64-bit. Otherwise FAIL. |
| BIOS Vendor | FIRM17 | **Required** Manual | Informational: Provides the BIOS vendor name. |
| Bluetooth | SENSOR-06 | **Required** Automated | Scans for and detects nearby Bluetooth*-enabled devices. The test expects to find a device "ubuntu0" and attempts to ping it. |
| Cellular_3G | WAN-04 | **Required** Automated | Search for the 3G interface and attempt to implement a cellular connection. |
| Cellular_SMS | WAN-03<br>WAN-04<br>WAN-05 | **Required** Automated | Search for a test phone number to send an SMS to verify and establish a cellular connection with the gateway. |
| Compact Flash | HW-33 | **Required** Manual | The gateway is not required to boot from a Compact Flash card.<br>1. Check for the presence of a Compact Flash connector on the board.<br>2. Plug in a Compact Flash card.<br>3. If the device displays, perform a read/write operation.<br>4. PASS if read/write is successful. Otherwise FAIL |

***continued...***

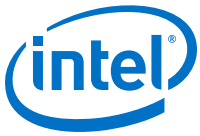


| Test Name | Test ID | Test Type | Description |
|---|---|---|---|
| CPU | PROC-01 | **Required** Automated | Find the presence of the correct gateway SKU: Intel® Quark™ or Atom™ processor. |
| DisplayPort | HW-08 | **Required** Manual | Look for a gateway display port. Required only on a gateway with an Intel® Atom™ processor as a commercial product. |
| ECC_Memory_Check | HW-02 | **Required** Manual | Manually check the presence of ECC memory on the board and in the BIOS configuration. |
| EFI_Shell | FIRM-16 | **Required** Manual | Check for the presence of EFI shell. |
| Edp | HW-08 | **Required** Manual | Check that the Electronic Data Processing display is connected to the display port. |
| Encrypted_Storage | OS-SEC-07 | **Required** Automated | Ensure encrypted storage can be set up. There is no error checking because a fail condition is not expected unless the gateway is missing the required executables. |
| ESATA | HW-34 | Optional Manual | Check for the presence of the ESATA device on the board. This is not applicable for an Intel® Quark™-based gateway. |
| ETH_Switch | WAN-01 | **Required** Automated | Test the Wi-Fi* connectivity of the Ethernet switches by connecting at least eight clients to AP, according to the *Intel® IoT Gateway Specification*. |
| Ethernet | WAN-01 | **Required** Automated | Confirm the Ethernet connection by pinging the gateway. |
| GPIO | SENSOR-03 | **Required** Automated | The test iterates through every GPIO port starting with first available port and attempts to enable one to user-space (sysfs). FAIL if a GPIO node can't be created. |
| GPS | SENSOR-10 | **Required** Manual | Test the presence of the GPS module. If present and configured, it can be used to test the exact location of the system for dimensionality. |
| Grsecurity | OS-SEC-02 | **Required** Automated | Verifies GRSecurity existence and functionality. |
| HDMI | HW-08 | **Required** Manual | Confirm the HDMI display is working by connecting the supported display interfaces into the display panel. |
| Headless Operation | FIRM-02 | **Required** Manual | 1. Check the ODM sheet to check to confirm we have all the connectors that go onto the board.<br>2. Connect all the cables to the respective ports onto the board.<br>3. Leaving out the display connector that is particular to the ODM board, ensure all the connectors are in place to be able to power on the system.<br>4. Power-on the board and confirm system boots by:<br>a. Manually checking if all the LEDs that indicate the board is powered on are blinking.<br>b. Using the IP address for the board, try to SSH into it from a Development Computer connected to the board to confirm that the system has successfully booted.<br>5. PASS if SSH connection is successful. Otherwise FAIL. |
| IPSec_VPN | OS-SEC-08 | **Required** Automated | Check for the presence of IPsec implementation in Wind River® Intelligent Device Platform XT 3. |
| Legacy Free Operation | FIRM-03 | **Required** Manual | 1. Check the ODM sheet to confirm the presence of all board connectors.<br>2. Connect all cables to the respective ports, including the applicable display cable and power cord.<br>3. Leave out the I/O connections such as the keyboard, mouse or any other input device required to operate the system. |

***continued...***

| Test Name | Test ID | Test Type | Description |
|---|---|---|---|
| | | | 4. Make sure a display monitor is connected to visually verify if the system has booted up.<br>5. Power on the board and let it boot.<br>6. View the LED lights that indicate that the board is powered on.<br>7. View the display monitor to see that system is booting without help on any input devices.<br>8. PASS if board can reach the terminal point without requiring the help of any input device. Otherwise FAIL. |
| LTP | OS-05 | **Required** Automated | Execute the Linux Test Project (LTP) test suite. |
| LUA | OS-APP-02 | **Required** Automated | Test if the LUA code can be compiled and run on the gateway. LUA is a multi-paradigm programming language that resides as a scripting package. |
| LVDS | HW-08 | **Required** Manual | Check if the LVDS cable is working by connecting the supported display interfaces to the display panel. |
| McAfee_Embedded_Control | OS-03 | **Required** Automated | Check the presence of the McAfee Embedded Control security feature and confirm it can be configured. |
| Memory Channel | HW-28 | **Required** Manual | Information: Provides the type of memory channel implemented. |
| Memory_Size | HW-01 | **Required** Automated | Verify the DRAM memory is at least 512 MB. |
| Memory Topology | HW-29 | **Required** Manual | Information: Provides the implemented memory topology. |
| Memory_Type | HW-02 | **Required** Manual | Check the memory type (SODIMM or memory down) and the number of memory channels on the board. |
| MQTT | OS-COMM-02 | **Required** Automated | Confirm the mosquitto message broker is present, and messaging using publish/subscribe model (MQTT protocol) is functional. |
| MSATA | FIRM-08 | Optional Manual | The boot service from MSATA is not applicable to Intel® Quark™-based gateways.<br>Check for the UEFI Boot services to boot from the MSATA and boot the system.<br>Only Boot Services is required. The OS provides drivers.<br>PASS if the board boots. Otherwise FAIL. |
| OMA-DM | OS-MANAGE-02 | **Required** Automated | Check the presence of OMA-DM manageability agent feature and confirm it can be configured. |
| OpenJDK | OS-APP-01 | **Required** Automated | Check the presence of OpenJDK & its version. |
| OpenSSL | OS-SEC-01 | **Required** Automated | Create public and private keys using openssl, sign a test file with the private key, and then recover the signed file data to confirm access. |
| Open POSIX Test Suite (OPTS) | OS-06 | **Required** Automated | Execute the Open POSIX Test Suite (OPTS). |
| OSGi | OS-APP-04 | **Required** Automated | Check the presence of OSGi feature. The ProSyst mBS Smart Home SDK development kit are included with OSGi bundle and provide a base for tailoring images for specific home device management platforms. Confirm it can be configured. |
| PWM | HW-26 | **Required** Automated | Smoke test. Confirms if PWM is accessible through user-space (sysfs). The test will fail if a PWM node can't be created. |
| Python | APP-01 | **Required** Automated | Check if Python is installed on the gateway and can be compiled and run on it. |

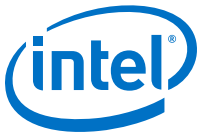
***continued...***



| Test Name | Test ID | Test Type | Description |
|---|---|---|---|
| RS232 | SENSOR-01 | **Required** Manual | Check the correct working and data transfer over RS232 serial interface. |
| RS485 | SENSOR-01 | **Required** Manual | Check the correct working and data transfer of RS-485 interface. |
| SATA | FIRM-08 | Optional Manual | Check for the presence of a SATA device. This is not applicable to Intel® Quark™-based Gateways.<br>Check for the UEFI Boot services to boot from the SATA and boot the system.<br>Only Boot Services is required. The OS provides drivers.<br>PASS if the board boots. Otherwise FAIL. |
| SD Card | HW-32 | Optional Manual | The gateway is not required to boot from an SD card.<br>1. Check for the presence of a SD card slot on the board.<br>2. Plug in a SD card.<br>3. If the device displays, perform a read/write operation.<br>4. PASS if read/write is successful. Otherwise FAIL. |
| Secure Boot | FIRM-14.1 | **Required** Manual | Confirm the system contains all software packages and features necessary to support secure boot capability. All gateways must be capable of Secure Boot. |
| Signed_Bootloader_Atom | OS-SEC-04 | **Required** Manual | Test the Bootloader application for the Intel® Atom™ processor-based Gateway. |
| Signed_Bootloader_Quark | OS-SEC-04 | **Required** Manual | Test the Intelligent Device Platform to check module presence, version to boot up the BIOS of the system by checking if there is certified data present. |
| Signed_Kernel_Atom | OS-SEC-05 | **Required** Manual | Check if the certificate data from the Bootloader has successfully passed to the kernel for an Intel® Atom™ processor-based Gateway. |
| Signed_Kernel_Quark | OS-SEC-05 | **Required** Manual | Check if the certificate data from the Bootloader has successfully passed to the kernel for an Intel® Quark™ processor-based Gateway. |
| SOC_Config | HW-03<br>HW-04<br>WAN-01<br>OS-SEC-07<br>PROC-01<br>HW-08 | **Required** Automated | Test the presence of system I/O interfaces, USB devices, Network IO interfaces, storage devices, CPU, and memory configuration. |
| SQLite3 | OS-APP-03 | **Required** Automated | Look at the log file to determine which version is installed. |
| Storage | HW-27 | **Required** Automated | Test storage testing by creating a random file, copying it to the gateway and moving it back. Use CMP and checksums to ensure both files are identical. |
| Touch Controller | HW-35 | Optional Manual | Test procedures are pending. |
| TPM | HW-07 | **Required** Automated | Check if the main board must host a Trusted Computing Group-compliant TPM v1.2 or ISO standard ISO/IEC 11889-1L2009 TPM. |
| TR-069 | OS-MANAGE-03 | **Required** Automated | Check the presence of TR-069 manageability agent feature and confirm it can be configured. |
| UEFI Boot | FIRM-14 | **Required** Manual | Confirm UEFI boot is supported in the BIOS.<br>1. Connect a device that contains a bootable image to the USB, MMC, or SATA connector.<br>2. Enter the BIOS utility by following the on-screen instructions at power-on. |

***continued...***

| Test Name | Test ID | Test Type | Description |
|---|---|---|---|
| | | | 3. In the BIOS utility, locate the 'Boot Option Priorities' option. The location will vary by BIOS vendor.<br>4. PASS if you see a boot option to boot the system using your boot-device as "UEFI: <Name of Boot-device>". Otherwise FAIL. |
| UEFI_Boot_Services | FIRM-08 | **Required** Manual | Check that the UEFI boot is supported in the BIOS for an USB, MMC/eMMC, and SATA/mSATA devices. |
| UEFI_Capsule_Update | FIRM-10 | **Required** Manual | Check if UEFI Capsule Update is enabled. |
| UEFI_Compliance | FIRM-05 | **Required** Manual | Check the if the BIOS in the system is Intel® IoT Gateway-compliant. UEFI version 2.3.1 errata C or later is required for compliance. |
| UEFI_Secure_Boot | FIRM-14 | **Required** Manual | Check the features of Secure Boot in BIOS. |
| USB | HW-08 | **Required** Automated | Check for the presence and to print USB information. |
| USB Hub | HW-31 | Optional Manual | Check the functionality of a USB hub. A USB hub is formed when a single USB port expands into several USB ports to make additional ports available for connecting devices.<br>1. Connect a USB hub into one of the available USB ports.<br>2. On the terminal, enter the command `lsusb`.<br>3. Look for the hub device name.<br>4. PASS if the device displays. Otherwise FAIL. |
| USB3_Flashdrive | HW-04 | **Required** Automated | This test is built from the USB test, but in this test, `parse-lsusb.py` looks specifically for a USB3 flash drive. |
| VGA | HW-08 | **Required** Manual | Check if the video graphics array connector is working by connecting it to the display panel. |
| Virtualization_Extensions | FIRM-04 | **Required** Automated | |
| Watchdog_Timer | HW-24 | **Required** Automated | Test if the Watchdog_Timer device file is present. |
| WiFi_AP_mode | SENSOR-05 | **Required** Automated | Determine if the Wind River Linux/Intelligent Device Platform XT 3 configures a system by default in AP mode / if its supported by the Wi-Fi module. |
| WiFi_Client | WAN-02 | **Required** Automated | Test the Wi-Fi connectivity for the gateway to behave as a client to enable connectivity. |



# Appendix D Third-Party Disclaimers

See the following pages for third-party disclaimers.

Cairo is free software.

Every source file in the implementation[*] of cairo is available to be redistributed and/or modified under the terms of either the GNU Lesser General Public License (LGPL) version 2.1 or the Mozilla Public License (MPL) version 1.1.  Some files are available under more liberal terms, but we believe that in all cases, each file may be used under either the LGPL or the MPL.

See the following files in this directory for the precise terms and conditions of either license:

    COPYING-LGPL-2.1
    COPYING-MPL-1.1

Please see each file in the implementation for copyright and licensing information, (in the opening comment of each file).

[*] The implementation of cairo is contained entirely within the "src" directory of the cairo source distribution. There are other components of the cairo source distribution (such as the "test", "util", and "perf") that are auxiliary to the library itself. None of the source code in these directories contributes to a build of the cairo library itself, (libcairo.so or cairo.dll or similar).

These auxiliary components are also free software, but may be under different license terms than cairo itself. For example, most of the test cases in the perf and test directories are made available under an MIT license to simplify any use of this code for reference purposes in using cairo itself. Other files might be available under the GNU General Public License (GPL), for example. Again, please see the COPYING file under each directory and the opening comment of each file for copyright and licensing information.

# GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999



[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

## Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is

modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

## GNU LESSER GENERAL PUBLIC LICENSE
## TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the

ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on

which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not

excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status

of all derivatives of our free software and of promoting the sharing
and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Lesser General Public
License as published by the Free Software Foundation; either
version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA  02110-1301  USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary.  Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

fontconfig/COPYING

Libgdiplus is available under the terms of the MIT X11 license.

The following is the MIT license, agreed upon by most contributors.
Copyright holders of new code should use this license statement where
possible. They may also add themselves to the list below.

used in advertising or publicity pertaining to distribution
of the software without specific prior written permission.
Silicon Graphics makes no representation about the suitability
of this software for any purpose. It is provided "as is"
without any express or implied warranty.
SILICON GRAPHICS DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS
SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY
AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL SILICON
GRAPHICS BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL
DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE,
DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE
OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH
THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright 1992 Network Computing Devices

Permission to use, copy, modify, distribute, and sell this software and its
documentation for any purpose is hereby granted without fee, provided that
the above copyright notice appear in all copies and that both that
copyright notice and this permission notice appear in supporting
documentation, and that the name of NCD. not be used in advertising or
publicity pertaining to distribution of the software without specific,
written prior permission. NCD. makes no representations about the
suitability of this software for any purpose. It is provided "as is"
without express or implied warranty.

NCD. DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL NCD.
BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION
OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN
CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Copyright 1991,1993 by Digital Equipment Corporation, Maynard, Massachusetts,
and Olivetti Research Limited, Cambridge, England.

All Rights Reserved

Permission to use, copy, modify, and distribute this software and its
documentation for any purpose and without fee is hereby granted,
provided that the above copyright notice appear in all copies and that
both that copyright notice and this permission notice appear in
supporting documentation, and that the names of Digital or Olivetti
not be used in advertising or publicity pertaining to distribution of the
software without specific, written prior permission.

DIGITAL AND OLIVETTI DISCLAIM ALL WARRANTIES WITH REGARD TO THIS
SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS, IN NO EVENT SHALL THEY BE LIABLE FOR ANY SPECIAL, INDIRECT OR
CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF
USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR
OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THIS SOFTWARE.

The Mono runtime is licensed under the terms of the GNU
Library General Public License, version 2.

The eglib directory is licensed under the terms of the MIT
X11 license and is a drop-in replacement for Mono's use of
glib 2.0 (which was LGPL).

The Boehm licensing information is in the libgc directory

The SGen Garbage Collector is under the terms of the MIT X11
license

The class libraries under mono/mcs are unless otherwise stated
under the MIT X11 license.

Open source Microsoft code is licensed under the original terms
which is either MS-PL for older components, or dual licensed
MS-PL/Apache2 licensed.

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991



[This is the first released version of the library GPL. It is
numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some
specially designated Free Software Foundation software, and to any
other libraries whose authors decide to use it. You can use it for
your libraries, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if

you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free

libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

GNU LIBRARY GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any

warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under

the scope of this License.

  3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library.  To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License.  (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.)  Do not make any other change in these notices.

  Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

  This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

  4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

  If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

  5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library".  Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

  However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library".  The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

  When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library.  The threshold for this to be true is not precisely defined by law.

  If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative

work.  (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License.  You must supply a copy of this License.  If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License.  Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library.  (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it.  However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies

the executable.

  It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system.  Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

  7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

    a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities.  This must be distributed under the terms of the Sections above.

    b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

  8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License.  Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License.  However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

  9. You are not required to accept this License, since you have not signed it.  However, nothing else grants you permission to modify or distribute the Library or its derivative works.  These actions are prohibited by law if you do not accept this License.  Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

  10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions.  You may not impose any further restrictions on the recipients' exercise of the rights granted herein.  You are not responsible for enforcing compliance by third parties to this License.

  11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License.  If you cannot

distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing

and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Appendix: How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Library General Public
License as published by the Free Software Foundation; either
version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Library General Public License for more details.

You should have received a copy of the GNU Library General Public

License along with this library; if not, write to the Free
Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary.  Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

The Code Project Open License (CPOL)

The Code Project Open License (CPOL) 1.02

This License governs Your use of the Work. This License is intended to allow developers to use the Source Code and Executable Files provided as part of the Work in any application in any form.

The main points subject to the terms of the License are:

Source Code and Executable Files can be used in commercial applications;
Source Code and Executable Files can be redistributed; and
Source Code can be modified to create derivative works.
No claim of suitability, guarantee, or any warranty whatsoever is provided. The software is provided "as-is".
The Article accompanying the Work may not be distributed or republished without the Author's consent

This License is entered between You, the individual or other entity reading or otherwise making use of the Work licensed pursuant to this License and the individual or other entity which offers the Work under the terms of this License ("Author").

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CODE PROJECT OPEN
LICENSE ("LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE
LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT
LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HEREIN, YOU ACCEPT AND AGREE TO BE
BOUND BY THE TERMS OF THIS LICENSE. THE AUTHOR GRANTS YOU THE RIGHTS CONTAINED HEREIN
IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS. IF YOU DO NOT
AGREE TO ACCEPT AND BE BOUND BY THE TERMS OF THIS LICENSE, YOU CANNOT MAKE ANY
USE OF THE WORK.

Definitions

"Articles" means, collectively, all articles written by Author which describes how the Source Code and Executable Files for the Work may be used by a user.
"Author" means the individual or entity that offers the Work under the terms of this License.
"Derivative Work" means a work based upon the Work or upon the

Work and other pre-existing works.
"Executable Files" refer to the executables, binary files, configuration and any required data files included in the Work.
"Publisher" means the provider of the website, magazine, CD-ROM, DVD or other medium from or by which the Work is obtained by You.
"Source Code" refers to the collection of source code and configuration files used to create the Executable Files.
"Standard Version" refers to such a Work if it has not been modified, or has been modified in accordance with the consent of the Author, such consent being in the full discretion of the Author.
"Work" refers to the collection of files distributed by the Publisher, including the Source Code, Executable Files, binaries, data files, documentation, whitepapers and the Articles.
"You" is you, an individual or entity wishing to use the Work and exercise your rights under this License.

Fair Use/Fair Use Rights. Nothing in this License is intended to reduce, limit, or restrict any rights arising from fair use, fair dealing, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

License Grant. Subject to the terms and conditions of this License, the Author hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

You may use the standard version of the Source Code or Executable Files in Your own applications.
You may apply bug fixes, portability fixes and other modifications obtained from the Public Domain or from the Author. A Work modified in such a way shall still be considered the standard version and will be subject to this License.
You may otherwise modify Your copy of this Work (excluding the Articles) in any way to create a Derivative Work, provided that You insert a prominent notice in each changed file stating how, when and where You changed that file.
You may distribute the standard version of the Executable Files and Source Code or Derivative Work in aggregate with other (possibly commercial) programs as part of a larger (possibly commercial) software distribution.
The Articles discussing the Work published in any form by the author may not be distributed or republished without the Author's consent. The author retains copyright to any such Articles. You may use the Executable Files and Source Code pursuant to this License but you may not repost or republish or otherwise distribute or make available the Articles, without the prior written consent of the Author.

Any subroutines or modules supplied by You and linked into the Source Code or Executable Files of this Work shall not be considered part of this Work and will not be subject to the terms of this License.

Patent License. Subject to the terms and conditions of this License,

each Author hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, import, and otherwise transfer the Work.

Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

- You agree not to remove any of the original copyright, patent, trademark, and attribution notices and associated disclaimers that may appear in the Source Code or Executable Files.
- You agree not to advertise or in any way imply that this Work is a product of Your own.
- The name of the Author may not be used to endorse or promote products derived from the Work without the prior written consent of the Author.
- You agree not to sell, lease, or rent any part of the Work. This does not restrict you from including the Work or any part of the Work inside a larger software distribution that itself is being sold. The Work by itself, though, cannot be sold, leased or rented.
- You may distribute the Executable Files and Source Code only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy of the Executable Files or Source Code You distribute and ensure that anyone receiving such Executable Files and Source Code agrees that the terms of this License apply to such Executable Files and/or Source Code. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute the Executable Files or Source Code with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License.
- You agree not to use the Work for illegal, immoral or improper purposes, or on pages containing illegal, immoral or improper material. The Work is subject to applicable export laws. You agree to comply with all such laws and regulations that may apply to the Work after Your receipt of the Work.

Representations, Warranties and Disclaimer. THIS WORK IS PROVIDED "AS IS", "WHERE IS" AND "AS AVAILABLE", WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES OR CONDITIONS OR GUARANTEES. YOU, THE USER, ASSUME ALL RISK IN ITS USE, INCLUDING COPYRIGHT INFRINGEMENT, PATENT INFRINGEMENT, SUITABILITY, ETC. AUTHOR EXPRESSLY DISCLAIMS ALL EXPRESS, IMPLIED OR STATUTORY WARRANTIES OR CONDITIONS, INCLUDING WITHOUT LIMITATION, WARRANTIES OR CONDITIONS OF MERCHANTABILITY, MERCHANTABLE QUALITY OR FITNESS FOR A PARTICULAR PURPOSE, OR ANY WARRANTY OF TITLE OR NON-INFRINGEMENT, OR THAT THE WORK (OR ANY PORTION THEREOF) IS CORRECT, USEFUL, BUG-FREE OR FREE OF VIRUSES. YOU MUST PASS THIS DISCLAIMER ON WHENEVER YOU DISTRIBUTE THE WORK OR DERIVATIVE

WORKS.

Indemnity. You agree to defend, indemnify and hold harmless the Author and the Publisher from and against any claims, suits, losses, damages, liabilities, costs, and expenses (including reasonable legal or attorneys' fees) resulting from or relating to any use of the Work by You.

Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL THE AUTHOR OR THE PUBLISHER BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK OR OTHERWISE, EVEN IF THE AUTHOR OR THE PUBLISHER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Termination.

<ol style="list-style-type: lower-alpha;">

This License and the rights granted hereunder will terminate automatically upon any breach by You of any term of this License. Individuals or entities who have received Derivative Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 6, 7, 8, 9, 10 and 11 will survive any termination of this License.

If You bring a copyright, trademark, patent or any other infringement claim against any contributor over infringements You claim are made by the Work, your License from such contributor to the Work ends automatically.

Subject to the above terms and conditions, this License is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, the Author reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

Publisher. The parties hereby confirm that the Publisher shall not, under any circumstances, be responsible for and shall not have any liability in respect of the subject matter of this License. The Publisher makes no warranty whatsoever in connection with the Work and shall not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. The Publisher reserves the right to cease making the Work available to You at any time without notice

Miscellaneous

This License shall be governed by the laws of the location of the head office of the Author or if the Author is an individual, the laws of location of the principal place of residence of the Author.

If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this License, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

This License constitutes the entire agreement between the parties with respect to the Work licensed herein. There are no understandings, agreements or representations with respect to the Work not specified herein. The Author shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Author and You.

```
/* png.h - header file for PNG reference library
 *
 * libpng version 1.6.13 - August 21, 2014
 * Copyright (c) 1998-2014 Glenn Randers-Pehrson
 * (Version 0.96 Copyright (c) 1996, 1997 Andreas Dilger)
 * (Version 0.88 Copyright (c) 1995, 1996 Guy Eric Schalnat, Group 42, Inc.)
 *
 * This code is released under the libpng license (See LICENSE, below)
 *
 * Authors and maintainers:
 *   libpng versions 0.71, May 1995, through 0.88, January 1996: Guy Schalnat
 *   libpng versions 0.89c, June 1996, through 0.96, May 1997: Andreas Dilger
 *   libpng versions 0.97, January 1998, through 1.6.13 - August 21, 2014: Glenn
 *   See also "Contributing Authors", below.
 *
 * Note about libpng version numbers:
 *
 *   Due to various miscommunications, unforeseen code incompatibilities
 *   and occasional factors outside the authors' control, version numbering
 *   on the library has not always been consistent and straightforward.
 *   The following table summarizes matters since version 0.89c, which was
 *   the first widely used release:
 *
 *    source               png.h  png.h  shared-lib
 *    version              string   int  version
 *    -------             ------ -----  ----------
 *    0.89c "1.0 beta 3"     0.89      89  1.0.89
 *    0.90  "1.0 beta 4"     0.90      90  0.90  [should have been 2.0.90]
 *    0.95  "1.0 beta 5"     0.95      95  0.95  [should have been 2.0.95]
 *    0.96  "1.0 beta 6"     0.96      96  0.96  [should have been 2.0.96]
 *    0.97b "1.00.97 beta 7" 1.00.97   97  1.0.1 [should have been 2.0.97]
 *    0.97c                  0.97      97  2.0.97
 *    0.98                   0.98      98  2.0.98
 *    0.99                   0.99      98  2.0.99
 *    0.99a-m                0.99      99  2.0.99
 *    1.00                   1.00     100  2.1.0 [100 should be 10000]
 *    1.0.0      (from here on, the    100  2.1.0 [100 should be 10000]
 *    1.0.1       png.h string is     10001  2.1.0
 *    1.0.1a-e     identical to the   10002  from here on, the shared library
 *    1.0.2       source version)     10002  is 2.V where V is the source code
 *    1.0.2a-b                        10003  version, except as noted.
 *    1.0.3                           10003
 *    1.0.3a-d                        10004
 *    1.0.4                           10004
 *    1.0.4a-f                        10005
 *    1.0.5 (+ 2 patches)             10005
 *    1.0.5a-d                        10006
 *    1.0.5e-r                        10100 (not source compatible)
 *    1.0.5s-v                        10006 (not binary compatible)
 *    1.0.6 (+ 3 patches)             10006 (still binary incompatible)
 *    1.0.6d-f                        10007 (still binary incompatible)
 *    1.0.6g                          10007
```

```
*    1.0.6h              10007  10.6h (testing xy.z so-numbering)
*    1.0.6i              10007  10.6i
*    1.0.6j              10007  2.1.0.6j (incompatible with 1.0.0)
*    1.0.7beta11-14      DLLNUM  10007  2.1.0.7beta11-14 (binary compatible)
*    1.0.7beta15-18       1    10007  2.1.0.7beta15-18 (binary compatible)
*    1.0.7rc1-2           1    10007  2.1.0.7rc1-2 (binary compatible)
*    1.0.7                1    10007  (still compatible)
*    1.0.8beta1-4         1    10008  2.1.0.8beta1-4
*    1.0.8rc1             1    10008  2.1.0.8rc1
*    1.0.8                1    10008  2.1.0.8
*    1.0.9beta1-6         1    10009  2.1.0.9beta1-6
*    1.0.9rc1             1    10009  2.1.0.9rc1
*    1.0.9beta7-10        1    10009  2.1.0.9beta7-10
*    1.0.9rc2             1    10009  2.1.0.9rc2
*    1.0.9                1    10009  2.1.0.9
*    1.0.10beta1          1    10010  2.1.0.10beta1
*    1.0.10rc1            1    10010  2.1.0.10rc1
*    1.0.10               1    10010  2.1.0.10
*    1.0.11beta1-3        1    10011  2.1.0.11beta1-3
*    1.0.11rc1            1    10011  2.1.0.11rc1
*    1.0.11               1    10011  2.1.0.11
*    1.0.12beta1-2        2    10012  2.1.0.12beta1-2
*    1.0.12rc1            2    10012  2.1.0.12rc1
*    1.0.12               2    10012  2.1.0.12
*    1.1.0a-f             -    10100  2.1.1.0a-f (branch abandoned)
*    1.2.0beta1-2         2    10200  2.1.2.0beta1-2
*    1.2.0beta3-5         3    10200  3.1.2.0beta3-5
*    1.2.0rc1             3    10200  3.1.2.0rc1
*    1.2.0                3    10200  3.1.2.0
*    1.2.1beta1-4         3    10201  3.1.2.1beta1-4
*    1.2.1rc1-2           3    10201  3.1.2.1rc1-2
*    1.2.1                3    10201  3.1.2.1
*    1.2.2beta1-6        12    10202  12.so.0.1.2.2beta1-6
*    1.0.13beta1         10    10013  10.so.0.1.0.13beta1
*    1.0.13rc1           10    10013  10.so.0.1.0.13rc1
*    1.2.2rc1            12    10202  12.so.0.1.2.2rc1
*    1.0.13              10    10013  10.so.0.1.0.13
*    1.2.2               12    10202  12.so.0.1.2.2
*    1.2.3rc1-6          12    10203  12.so.0.1.2.3rc1-6
*    1.2.3               12    10203  12.so.0.1.2.3
*    1.2.4beta1-3        13    10204  12.so.0.1.2.4beta1-3
*    1.0.14rc1           13    10014  10.so.0.1.0.14rc1
*    1.2.4rc1            13    10204  12.so.0.1.2.4rc1
*    1.0.14              10    10014  10.so.0.1.0.14
*    1.2.4               13    10204  12.so.0.1.2.4
*    1.2.5beta1-2        13    10205  12.so.0.1.2.5beta1-2
*    1.0.15rc1-3         10    10015  10.so.0.1.0.15rc1-3
*    1.2.5rc1-3          13    10205  12.so.0.1.2.5rc1-3
*    1.0.15              10    10015  10.so.0.1.0.15
*    1.2.5               13    10205  12.so.0.1.2.5
*    1.2.6beta1-4        13    10206  12.so.0.1.2.6beta1-4
*    1.0.16              10    10016  10.so.0.1.0.16
*    1.2.6               13    10206  12.so.0.1.2.6
*    1.2.7beta1-2        13    10207  12.so.0.1.2.7beta1-2
```

```
*    1.0.17rc1            10    10017  12.so.0.1.0.17rc1
*    1.2.7rc1             13    10207  12.so.0.1.2.7rc1
*    1.0.17               10    10017  12.so.0.1.0.17
*    1.2.7                13    10207  12.so.0.1.2.7
*    1.2.8beta1-5         13    10208  12.so.0.1.2.8beta1-5
*    1.0.18rc1-5          10    10018  12.so.0.1.0.18rc1-5
*    1.2.8rc1-5           13    10208  12.so.0.1.2.8rc1-5
*    1.0.18               10    10018  12.so.0.1.0.18
*    1.2.8                13    10208  12.so.0.1.2.8
*    1.2.9beta1-3         13    10209  12.so.0.1.2.9beta1-3
*    1.2.9beta4-11        13    10209  12.so.0.9[.0]
*    1.2.9rc1             13    10209  12.so.0.9[.0]
*    1.2.9                13    10209  12.so.0.9[.0]
*    1.2.10beta1-7        13    10210  12.so.0.10[.0]
*    1.2.10rc1-2          13    10210  12.so.0.10[.0]
*    1.2.10               13    10210  12.so.0.10[.0]
*    1.4.0beta1-5         14    10400  14.so.0.0[.0]
*    1.2.11beta1-4        13    10211  12.so.0.11[.0]
*    1.4.0beta7-8         14    10400  14.so.0.0[.0]
*    1.2.11               13    10211  12.so.0.11[.0]
*    1.2.12               13    10212  12.so.0.12[.0]
*    1.4.0beta9-14        14    10400  14.so.0.0[.0]
*    1.2.13               13    10213  12.so.0.13[.0]
*    1.4.0beta15-36       14    10400  14.so.0.0[.0]
*    1.4.0beta37-87       14    10400  14.so.14.0[.0]
*    1.4.0rc01            14    10400  14.so.14.0[.0]
*    1.4.0beta88-109      14    10400  14.so.14.0[.0]
*    1.4.0rc02-08         14    10400  14.so.14.0[.0]
*    1.4.0                14    10400  14.so.14.0[.0]
*    1.4.1beta01-03       14    10401  14.so.14.1[.0]
*    1.4.1rc01            14    10401  14.so.14.1[.0]
*    1.4.1beta04-12       14    10401  14.so.14.1[.0]
*    1.4.1                14    10401  14.so.14.1[.0]
*    1.4.2                14    10402  14.so.14.2[.0]
*    1.4.3                14    10403  14.so.14.3[.0]
*    1.4.4                14    10404  14.so.14.4[.0]
*    1.5.0beta01-58       15    10500  15.so.15.0[.0]
*    1.5.0rc01-07         15    10500  15.so.15.0[.0]
*    1.5.0                15    10500  15.so.15.0[.0]
*    1.5.1beta01-11       15    10501  15.so.15.1[.0]
*    1.5.1rc01-02         15    10501  15.so.15.1[.0]
*    1.5.1                15    10501  15.so.15.1[.0]
*    1.5.2beta01-03       15    10502  15.so.15.2[.0]
*    1.5.2rc01-03         15    10502  15.so.15.2[.0]
*    1.5.2                15    10502  15.so.15.2[.0]
*    1.5.3beta01-10       15    10503  15.so.15.3[.0]
*    1.5.3rc01-02         15    10503  15.so.15.3[.0]
*    1.5.3beta11          15    10503  15.so.15.3[.0]
*    1.5.3 [omitted]
*    1.5.4beta01-08       15    10504  15.so.15.4[.0]
*    1.5.4rc01            15    10504  15.so.15.4[.0]
*    1.5.4                15    10504  15.so.15.4[.0]
*    1.5.5beta01-08       15    10505  15.so.15.5[.0]
*    1.5.5rc01            15    10505  15.so.15.5[.0]
```

```
 *   1.5.5               15    10505  15.so.15.5[.0]
 *   1.5.6beta01-07        15    10506  15.so.15.6[.0]
 *   1.5.6rc01-03          15    10506  15.so.15.6[.0]
 *   1.5.6               15    10506  15.so.15.6[.0]
 *   1.5.7beta01-05        15    10507  15.so.15.7[.0]
 *   1.5.7rc01-03          15    10507  15.so.15.7[.0]
 *   1.5.7               15    10507  15.so.15.7[.0]
 *   1.6.0beta01-40        16    10600  16.so.16.0[.0]
 *   1.6.0rc01-08          16    10600  16.so.16.0[.0]
 *   1.6.0               16    10600  16.so.16.0[.0]
 *   1.6.1beta01-09        16    10601  16.so.16.1[.0]
 *   1.6.1rc01             16    10601  16.so.16.1[.0]
 *   1.6.1               16    10601  16.so.16.1[.0]
 *   1.6.2beta01           16    10602  16.so.16.2[.0]
 *   1.6.2rc01-06          16    10602  16.so.16.2[.0]
 *   1.6.2               16    10602  16.so.16.2[.0]
 *   1.6.3beta01-11        16    10603  16.so.16.3[.0]
 *   1.6.3rc01             16    10603  16.so.16.3[.0]
 *   1.6.3               16    10603  16.so.16.3[.0]
 *   1.6.4beta01-02        16    10604  16.so.16.4[.0]
 *   1.6.4rc01             16    10604  16.so.16.4[.0]
 *   1.6.4               16    10604  16.so.16.4[.0]
 *   1.6.5               16    10605  16.so.16.5[.0]
 *   1.6.6               16    10606  16.so.16.6[.0]
 *   1.6.7beta01-04        16    10607  16.so.16.7[.0]
 *   1.6.7rc01-03          16    10607  16.so.16.7[.0]
 *   1.6.7               16    10607  16.so.16.7[.0]
 *   1.6.8beta01-02        16    10608  16.so.16.8[.0]
 *   1.6.8rc01-02          16    10608  16.so.16.8[.0]
 *   1.6.8               16    10608  16.so.16.8[.0]
 *   1.6.9beta01-04        16    10609  16.so.16.9[.0]
 *   1.6.9rc01-02          16    10609  16.so.16.9[.0]
 *   1.6.9               16    10609  16.so.16.9[.0]
 *   1.6.10beta01-03       16    10610  16.so.16.10[.0]
 *   1.6.10rc01-03         16    10610  16.so.16.10[.0]
 *   1.6.10              16    10610  16.so.16.10[.0]
 *   1.6.11beta01-06       16    10611  16.so.16.11[.0]
 *   1.6.11rc01-02         16    10611  16.so.16.11[.0]
 *   1.6.11              16    10611  16.so.16.11[.0]
 *   1.6.12rc01-03         16    10612  16.so.16.12[.0]
 *   1.6.12              16    10612  16.so.16.12[.0]
 *   1.6.13beta01-04       16    10613  16.so.16.13[.0]
 *   1.6.13rc01-02         16    10613  16.so.16.13[.0]
 *   1.6.13              16    10613  16.so.16.13[.0]
 *
 *   Henceforth the source version will match the shared-library major
 *   and minor numbers; the shared-library major version number will be
 *   used for changes in backward compatibility, as it is intended.  The
 *   PNG_LIBPNG_VER macro, which is not used within libpng but is available
 *   for applications, is an unsigned integer of the form xyyzz corresponding
 *   to the source version x.y.z (leading zeros in y and z).  Beta versions
 *   were given the previous public release number plus a letter, until
 *   version 1.0.6j; from then on they were given the upcoming public
 *   release number plus "betaNN" or "rcNN".
```

```
 *
 *   Binary incompatibility exists only when applications make direct access
 *   to the info_ptr or png_ptr members through png.h, and the compiled
 *   application is loaded with a different version of the library.
 *
 *   DLLNUM will change each time there are forward or backward changes
 *   in binary compatibility (e.g., when a new feature is added).
 *
 * See libpng-manual.txt or libpng.3 for more information.  The PNG
 * specification is available as a W3C Recommendation and as an ISO
 * Specification, <http://www.w3.org/TR/2003/REC-PNG-20031110/
 */

/*
 * COPYRIGHT NOTICE, DISCLAIMER, and LICENSE:
 *
 * If you modify libpng you may insert additional notices immediately following
 * this sentence.
 *
 * This code is released under the libpng license.
 *
 * libpng versions 1.2.6, August 15, 2004, through 1.6.13, August 21, 2014, are
 * Copyright (c) 2004, 2006-2014 Glenn Randers-Pehrson, and are
 * distributed according to the same disclaimer and license as libpng-1.2.5
 * with the following individual added to the list of Contributing Authors:
 *
 *    Cosmin Truta
 *
 * libpng versions 1.0.7, July 1, 2000, through 1.2.5, October 3, 2002, are
 * Copyright (c) 2000-2002 Glenn Randers-Pehrson, and are
 * distributed according to the same disclaimer and license as libpng-1.0.6
 * with the following individuals added to the list of Contributing Authors:
 *
 *    Simon-Pierre Cadieux
 *    Eric S. Raymond
 *    Gilles Vollant
 *
 * and with the following additions to the disclaimer:
 *
 *    There is no warranty against interference with your enjoyment of the
 *    library or against infringement.  There is no warranty that our
 *    efforts or the library will fulfill any of your particular purposes
 *    or needs.  This library is provided with all faults, and the entire
 *    risk of satisfactory quality, performance, accuracy, and effort is with
 *    the user.
 *
 * libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are
 * Copyright (c) 1998, 1999, 2000 Glenn Randers-Pehrson, and are
 * distributed according to the same disclaimer and license as libpng-0.96,
 * with the following individuals added to the list of Contributing Authors:
 *
 *    Tom Lane
 *    Glenn Randers-Pehrson
 *    Willem van Schaik
```

```
 *
 * libpng versions 0.89, June 1996, through 0.96, May 1997, are
 * Copyright (c) 1996, 1997 Andreas Dilger
 * Distributed according to the same disclaimer and license as libpng-0.88,
 * with the following individuals added to the list of Contributing Authors:
 *
 *    John Bowler
 *    Kevin Bracey
 *    Sam Bushell
 *    Magnus Holmgren
 *    Greg Roelofs
 *    Tom Tanner
 *
 * libpng versions 0.5, May 1995, through 0.88, January 1996, are
 * Copyright (c) 1995, 1996 Guy Eric Schalnat, Group 42, Inc.
 *
 * For the purposes of this copyright and license, "Contributing Authors"
 * is defined as the following set of individuals:
 *
 *    Andreas Dilger
 *    Dave Martindale
 *    Guy Eric Schalnat
 *    Paul Schmidt
 *    Tim Wegner
 *
 * The PNG Reference Library is supplied "AS IS".  The Contributing Authors
 * and Group 42, Inc. disclaim all warranties, expressed or implied,
 * including, without limitation, the warranties of merchantability and of
 * fitness for any purpose.  The Contributing Authors and Group 42, Inc.
 * assume no liability for direct, indirect, incidental, special, exemplary,
 * or consequential damages, which may result from the use of the PNG
 * Reference Library, even if advised of the possibility of such damage.
 *
 * Permission is hereby granted to use, copy, modify, and distribute this
 * source code, or portions hereof, for any purpose, without fee, subject
 * to the following restrictions:
 *
 *   1. The origin of this source code must not be misrepresented.
 *
 *   2. Altered versions must be plainly marked as such and must not
 *      be misrepresented as being the original source.
 *
 *   3. This Copyright notice may not be removed or altered from
 *      any source or altered source distribution.
 *
 * The Contributing Authors and Group 42, Inc. specifically permit, without
 * fee, and encourage the use of this source code as a component to
 * supporting the PNG file format in commercial products.  If you use this
 * source code in a product, acknowledgment is not required but would be
 * appreciated.
 */

/*
 * A "png_get_copyright" function is available, for convenient use in "about"
```

```
 * boxes and the like:
 *
 *     printf("%s", png_get_copyright(NULL));
 *
 * Also, the PNG logo (in PNG format, of course) is supplied in the
 * files "pngbar.png" and "pngbar.jpg (88x31) and "pngnow.png" (98x31).
 */

/*
 * Libpng is OSI Certified Open Source Software.  OSI Certified is a
 * certification mark of the Open Source Initiative.
 */

/*
 * The contributing authors would like to thank all those who helped
 * with testing, bug fixes, and patience.  This wouldn't have been
 * possible without all of you.
 *
 * Thanks to Frank J. T. Wojcik for helping with the documentation.
 */

/*
 * Y2K compliance in libpng:
 * =========================
 *
 *    August 21, 2014
 *
 *    Since the PNG Development group is an ad-hoc body, we can't make
 *    an official declaration.
 *
 *    This is your unofficial assurance that libpng from version 0.71 and
 *    upward through 1.6.13 are Y2K compliant.  It is my belief that
 *    earlier versions were also Y2K compliant.
 *
 *    Libpng only has two year fields.  One is a 2-byte unsigned integer
 *    that will hold years up to 65535.  The other, which is deprecated,
 *    holds the date in text format, and will hold years up to 9999.
 *
 *    The integer is
 *        "png_uint_16 year" in png_time_struct.
 *
 *    The string is
 *        "char time_buffer[29]" in png_struct.  This is no longer used
 *    in libpng-1.6.x and will be removed from libpng-1.7.0.
 *
 *    There are seven time-related functions:
 *        png.c: png_convert_to_rfc_1123_buffer() in png.c
 *          (formerly png_convert_to_rfc_1123() prior to libpng-1.5.x and
 *          png_convert_to_rfc_1152() in error prior to libpng-0.98)
 *        png_convert_from_struct_tm() in pngwrite.c, called in pngwrite.c
 *        png_convert_from_time_t() in pngwrite.c
 *        png_get_tIME() in pngget.c
 *        png_handle_tIME() in pngrutil.c, called in pngread.c
 *        png_set_tIME() in pngset.c
```

```
 *      png_write_tIME() in pngwutil.c, called in pngwrite.c
 *
 *    All handle dates properly in a Y2K environment.  The
 *    png_convert_from_time_t() function calls gmtime() to convert from system
 *    clock time, which returns (year - 1900), which we properly convert to
 *    the full 4-digit year.  There is a possibility that libpng applications
 *    are not passing 4-digit years into the png_convert_to_rfc_1123_buffer()
 *    function, or that they are incorrectly passing only a 2-digit year
 *    instead of "year - 1900" into the png_convert_from_struct_tm() function,
 *    but this is not under our control.  The libpng documentation has always
 *    stated that it works with 4-digit years, and the APIs have been
 *    documented as such.
 *
 *    The tIME chunk itself is also Y2K compliant.  It uses a 2-byte unsigned
 *    integer to hold the year, and can hold years as large as 65535.
 *
 *    zlib, upon which libpng depends, is also Y2K compliant.  It contains
 *    no date-related code.
 *
 *      Glenn Randers-Pehrson
 *      libpng maintainer
 *      PNG Development Group
 */

#ifndef PNG_H
#define PNG_H

/* This is not the place to learn how to use libpng. The file libpng-manual.txt
 * describes how to use libpng, and the file example.c summarizes it
 * with some code on which to build.  This file is useful for looking
 * at the actual function definitions and structure components.  If that
 * file has been stripped from your copy of libpng, you can find it at
 * <http://www.libpng.org/pub/png/libpng-manual.txt>
 *
 * If you just need to read a PNG file and don't want to read the documentation
 * skip to the end of this file and read the section entitled 'simplified API'.
 */

/* Version information for png.h - this should match the version in png.c */
#define PNG_LIBPNG_VER_STRING "1.6.13"
#define PNG_HEADER_VERSION_STRING \
    " libpng version 1.6.13 - August 21, 2014\n"

#define PNG_LIBPNG_VER_SONUM   16
#define PNG_LIBPNG_VER_DLLNUM  16

/* These should match the first 3 components of PNG_LIBPNG_VER_STRING: */
#define PNG_LIBPNG_VER_MAJOR   1
#define PNG_LIBPNG_VER_MINOR   6
#define PNG_LIBPNG_VER_RELEASE 13

/* This should match the numeric part of the final component of
 * PNG_LIBPNG_VER_STRING, omitting any leading zero:
 */
```

```
#define PNG_LIBPNG_VER_BUILD  0

/* Release Status */
#define PNG_LIBPNG_BUILD_ALPHA    1
#define PNG_LIBPNG_BUILD_BETA     2
#define PNG_LIBPNG_BUILD_RC       3
#define PNG_LIBPNG_BUILD_STABLE   4
#define PNG_LIBPNG_BUILD_RELEASE_STATUS_MASK 7

/* Release-Specific Flags */
#define PNG_LIBPNG_BUILD_PATCH    8 /* Can be OR'ed with
                                       PNG_LIBPNG_BUILD_STABLE only */
#define PNG_LIBPNG_BUILD_PRIVATE 16 /* Cannot be OR'ed with
                                       PNG_LIBPNG_BUILD_SPECIAL */
#define PNG_LIBPNG_BUILD_SPECIAL 32 /* Cannot be OR'ed with
                                       PNG_LIBPNG_BUILD_PRIVATE */

#define PNG_LIBPNG_BUILD_BASE_TYPE PNG_LIBPNG_BUILD_STABLE

/* Careful here.  At one time, Guy wanted to use 082, but that would be octal.
 * We must not include leading zeros.
 * Versions 0.7 through 1.0.0 were in the range 0 to 100 here (only
 * version 1.0.0 was mis-numbered 100 instead of 10000).  From
 * version 1.0.1 it's    xxyyzz, where x=major, y=minor, z=release
 */
#define PNG_LIBPNG_VER 10613 /* 1.6.13 */

/* Library configuration: these options cannot be changed after
 * the library has been built.
 */
#ifndef PNGLCONF_H
    /* If pnglibconf.h is missing, you can
     * copy scripts/pnglibconf.h.prebuilt to pnglibconf.h
     */
#   include "pnglibconf.h"
#endif

#ifndef PNG_VERSION_INFO_ONLY
   /* Machine specific configuration. */
#  include "pngconf.h"
#endif

/*
 * Added at libpng-1.2.8
 *
 * Ref MSDN: Private as priority over Special
 * VS_FF_PRIVATEBUILD File *was not* built using standard release
 * procedures. If this value is given, the StringFileInfo block must
 * contain a PrivateBuild string.
 *
 * VS_FF_SPECIALBUILD File *was* built by the original company using
 * standard release procedures but is a variation of the standard
 * file of the same version number. If this value is given, the
```

```
 * StringFileInfo block must contain a SpecialBuild string.
 */

#ifdef PNG_USER_PRIVATEBUILD /* From pnglibconf.h */
#  define PNG_LIBPNG_BUILD_TYPE \
      (PNG_LIBPNG_BUILD_BASE_TYPE | PNG_LIBPNG_BUILD_PRIVATE)
#else
#  ifdef PNG_LIBPNG_SPECIALBUILD
#    define PNG_LIBPNG_BUILD_TYPE \
        (PNG_LIBPNG_BUILD_BASE_TYPE | PNG_LIBPNG_BUILD_SPECIAL)
#  else
#    define PNG_LIBPNG_BUILD_TYPE (PNG_LIBPNG_BUILD_BASE_TYPE)
#  endif
#endif

#ifndef PNG_VERSION_INFO_ONLY

/* Inhibit C++ name-mangling for libpng functions but not for system calls. */
#ifdef __cplusplus
extern "C" {
#endif /* __cplusplus */

/* Version information for C files, stored in png.c.  This had better match
 * the version above.
 */
#define png_libpng_ver png_get_header_ver(NULL)

/* This file is arranged in several sections:
 *
 * 1. Any configuration options that can be specified by for the application
 *    code when it is built.  (Build time configuration is in pnglibconf.h)
 * 2. Type definitions (base types are defined in pngconf.h), structure
 *    definitions.
 * 3. Exported library functions.
 * 4. Simplified API.
 *
 * The library source code has additional files (principally pngpriv.h) that
 * allow configuration of the library.
 */
/* Section 1: run time configuration
 * See pnglibconf.h for build time configuration
 *
 * Run time configuration allows the application to choose between
 * implementations of certain arithmetic APIs.  The default is set
 * at build time and recorded in pnglibconf.h, but it is safe to
 * override these (and only these) settings.  Note that this won't
 * change what the library does, only application code, and the
 * settings can (and probably should) be made on a per-file basis
 * by setting the #defines before including png.h
 *
 * Use macros to read integers from PNG data or use the exported
 * functions?
 *   PNG_USE_READ_MACROS: use the macros (see below)  Note that
 *     the macros evaluate their argument multiple times.
```

```
 *   PNG_NO_USE_READ_MACROS: call the relevant library function.
 *
 * Use the alternative algorithm for compositing alpha samples that
 * does not use division?
 *   PNG_READ_COMPOSITE_NODIV_SUPPORTED: use the 'no division'
 *      algorithm.
 *   PNG_NO_READ_COMPOSITE_NODIV: use the 'division' algorithm.
 *
 * How to handle benign errors if PNG_ALLOW_BENIGN_ERRORS is
 * false?
 *   PNG_ALLOW_BENIGN_ERRORS: map calls to the benign error
 *      APIs to png_warning.
 * Otherwise the calls are mapped to png_error.
 */

/* Section 2: type definitions, including structures and compile time
 * constants.
 * See pngconf.h for base types that vary by machine/system
 */

/* This triggers a compiler error in png.c, if png.c and png.h
 * do not agree upon the version number.
 */
typedef char* png_libpng_version_1_6_13;

/* Basic control structions.  Read libpng-manual.txt or libpng.3 for more info.
 *
 * png_struct is the cache of information used while reading or writing a single
 * PNG file.  One of these is always required, although the simplified API
 * (below) hides the creation and destruction of it.
 */
typedef struct png_struct_def png_struct;
typedef const png_struct * png_const_structp;
typedef png_struct * png_structp;
typedef png_struct * * png_structpp;

/* png_info contains information read from or to be written to a PNG file.  One
 * or more of these must exist while reading or creating a PNG file.  The
 * information is not used by libpng during read but is used to control what
 * gets written when a PNG file is created.  "png_get_" function calls read
 * information during read and "png_set_" functions calls write information
 * when creating a PNG.
 * been moved into a separate header file that is not accessible to
 * applications.  Read libpng-manual.txt or libpng.3 for more info.
 */
typedef struct png_info_def png_info;
typedef png_info * png_infop;
typedef const png_info * png_const_infop;
typedef png_info * * png_infopp;

/* Types with names ending 'p' are pointer types.  The corresponding types with
 * names ending 'rp' are identical pointer types except that the pointer is
 * marked 'restrict', which means that it is the only pointer to the object
 * passed to the function.  Applications should not use the 'restrict' types;
```

```
 * it is always valid to pass 'p' to a pointer with a function argument of the
 * corresponding 'rp' type.  Different compilers have different rules with
 * regard to type matching in the presence of 'restrict'.  For backward
 * compatibility libpng callbacks never have 'restrict' in their parameters and,
 * consequentially, writing portable application code is extremely difficult if
 * an attempt is made to use 'restrict'.
 */
typedef png_struct * PNG_RESTRICT png_structrp;
typedef const png_struct * PNG_RESTRICT png_const_structrp;
typedef png_info * PNG_RESTRICT png_inforp;
typedef const png_info * PNG_RESTRICT png_const_inforp;

/* Three color definitions.  The order of the red, green, and blue, (and the
 * exact size) is not important, although the size of the fields need to
 * be png_byte or png_uint_16 (as defined below).
 */
typedef struct png_color_struct
{
   png_byte red;
   png_byte green;
   png_byte blue;
} png_color;
typedef png_color * png_colorp;
typedef const png_color * png_const_colorp;
typedef png_color * * png_colorpp;

typedef struct png_color_16_struct
{
   png_byte index;    /* used for palette files */
   png_uint_16 red;   /* for use in red green blue files */
   png_uint_16 green;
   png_uint_16 blue;
   png_uint_16 gray;  /* for use in grayscale files */
} png_color_16;
typedef png_color_16 * png_color_16p;
typedef const png_color_16 * png_const_color_16p;
typedef png_color_16 * * png_color_16pp;

typedef struct png_color_8_struct
{
   png_byte red;   /* for use in red green blue files */
   png_byte green;
   png_byte blue;
   png_byte gray;  /* for use in grayscale files */
   png_byte alpha; /* for alpha channel files */
} png_color_8;
typedef png_color_8 * png_color_8p;
typedef const png_color_8 * png_const_color_8p;
typedef png_color_8 * * png_color_8pp;

/*
 * The following two structures are used for the in-core representation
 * of sPLT chunks.
 */
```

```
typedef struct png_sPLT_entry_struct
{
   png_uint_16 red;
   png_uint_16 green;
   png_uint_16 blue;
   png_uint_16 alpha;
   png_uint_16 frequency;
} png_sPLT_entry;
typedef png_sPLT_entry * png_sPLT_entryp;
typedef const png_sPLT_entry * png_const_sPLT_entryp;
typedef png_sPLT_entry * * png_sPLT_entrypp;

/*  When the depth of the sPLT palette is 8 bits, the color and alpha samples
 *  occupy the LSB of their respective members, and the MSB of each member
 *  is zero-filled.  The frequency member always occupies the full 16 bits.
 */

typedef struct png_sPLT_struct
{
   png_charp name;           /* palette name */
   png_byte depth;           /* depth of palette samples */
   png_sPLT_entryp entries;  /* palette entries */
   png_int_32 nentries;      /* number of palette entries */
} png_sPLT_t;
typedef png_sPLT_t * png_sPLT_tp;
typedef const png_sPLT_t * png_const_sPLT_tp;
typedef png_sPLT_t * * png_sPLT_tpp;

#ifdef PNG_TEXT_SUPPORTED
/* png_text holds the contents of a text/ztxt/itxt chunk in a PNG file,
 * and whether that contents is compressed or not.  The "key" field
 * points to a regular zero-terminated C string.  The "text" fields can be a
 * regular C string, an empty string, or a NULL pointer.
 * However, the structure returned by png_get_text() will always contain
 * the "text" field as a regular zero-terminated C string (possibly
 * empty), never a NULL pointer, so it can be safely used in printf() and
 * other string-handling functions.  Note that the "itxt_length", "lang", and
 * "lang_key" members of the structure only exist when the library is built
 * with iTXt chunk support.  Prior to libpng-1.4.0 the library was built by
 * default without iTXt support. Also note that when iTXt *is* supported,
 * the "lang" and "lang_key" fields contain NULL pointers when the
 * "compression" field contains * PNG_TEXT_COMPRESSION_NONE or
 * PNG_TEXT_COMPRESSION_zTXt. Note that the "compression value" is not the
 * same as what appears in the PNG tEXt/zTXt/iTXt chunk's "compression flag"
 * which is always 0 or 1, or its "compression method" which is always 0.
 */
typedef struct png_text_struct
{
   int  compression;       /* compression value:
                             -1: tEXt, none
                              0: zTXt, deflate
                              1: iTXt, none
                              2: iTXt, deflate  */
   png_charp key;          /* keyword, 1-79 character description of "text" */
```

```
   png_charp text;         /* comment, may be an empty string (ie "")
                           or a NULL pointer */
   png_size_t text_length; /* length of the text string */
   png_size_t itxt_length; /* length of the itxt string */
   png_charp lang;         /* language code, 0-79 characters
                           or a NULL pointer */
   png_charp lang_key;     /* keyword translated UTF-8 string, 0 or more
                           chars or a NULL pointer */
} png_text;
typedef png_text * png_textp;
typedef const png_text * png_const_textp;
typedef png_text * * png_textpp;
#endif

/* Supported compression types for text in PNG files (tEXt, and zTXt).
 * The values of the PNG_TEXT_COMPRESSION_ defines should NOT be changed. */
#define PNG_TEXT_COMPRESSION_NONE_WR -3
#define PNG_TEXT_COMPRESSION_zTXt_WR -2
#define PNG_TEXT_COMPRESSION_NONE    -1
#define PNG_TEXT_COMPRESSION_zTXt     0
#define PNG_ITXT_COMPRESSION_NONE     1
#define PNG_ITXT_COMPRESSION_zTXt     2
#define PNG_TEXT_COMPRESSION_LAST     3  /* Not a valid value */

/* png_time is a way to hold the time in an machine independent way.
 * Two conversions are provided, both from time_t and struct tm.  There
 * is no portable way to convert to either of these structures, as far
 * as I know.  If you know of a portable way, send it to me.  As a side
 * note - PNG has always been Year 2000 compliant!
 */
typedef struct png_time_struct
{
   png_uint_16 year; /* full year, as in, 1995 */
   png_byte month;   /* month of year, 1 - 12 */
   png_byte day;     /* day of month, 1 - 31 */
   png_byte hour;    /* hour of day, 0 - 23 */
   png_byte minute;  /* minute of hour, 0 - 59 */
   png_byte second;  /* second of minute, 0 - 60 (for leap seconds) */
} png_time;
typedef png_time * png_timep;
typedef const png_time * png_const_timep;
typedef png_time * * png_timepp;

#if defined(PNG_STORE_UNKNOWN_CHUNKS_SUPPORTED) ||\
   defined(PNG_USER_CHUNKS_SUPPORTED)
/* png_unknown_chunk is a structure to hold queued chunks for which there is
 * no specific support.  The idea is that we can use this to queue
 * up private chunks for output even though the library doesn't actually
 * know about their semantics.
 *
 * The data in the structure is set by libpng on read and used on write.
 */
typedef struct png_unknown_chunk_t
{
```

```
   png_byte name[5]; /* Textual chunk name with '\0' terminator */
   png_byte *data;   /* Data, should not be modified on read! */
   png_size_t size;

   /* On write 'location' must be set using the flag values listed below.
    * Notice that on read it is set by libpng however the values stored have
    * more bits set than are listed below.  Always treat the value as a
    * bitmask.  On write set only one bit - setting multiple bits may cause the
    * chunk to be written in multiple places.
    */
   png_byte location; /* mode of operation at read time */
}
png_unknown_chunk;

typedef png_unknown_chunk * png_unknown_chunkp;
typedef const png_unknown_chunk * png_const_unknown_chunkp;
typedef png_unknown_chunk * * png_unknown_chunkpp;
#endif

/* Flag values for the unknown chunk location byte. */
#define PNG_HAVE_IHDR  0x01
#define PNG_HAVE_PLTE  0x02
#define PNG_AFTER_IDAT 0x08

/* Maximum positive integer used in PNG is (2^31)-1 */
#define PNG_UINT_31_MAX ((png_uint_32)0x7fffffffL)
#define PNG_UINT_32_MAX ((png_uint_32)(-1))
#define PNG_SIZE_MAX ((png_size_t)(-1))

/* These are constants for fixed point values encoded in the
 * PNG specification manner (x100000)
 */
#define PNG_FP_1    100000
#define PNG_FP_HALF  50000
#define PNG_FP_MAX  ((png_fixed_point)0x7fffffffL)
#define PNG_FP_MIN  (-PNG_FP_MAX)

/* These describe the color_type field in png_info. */
/* color type masks */
#define PNG_COLOR_MASK_PALETTE    1
#define PNG_COLOR_MASK_COLOR      2
#define PNG_COLOR_MASK_ALPHA      4

/* color types.  Note that not all combinations are legal */
#define PNG_COLOR_TYPE_GRAY 0
#define PNG_COLOR_TYPE_PALETTE  (PNG_COLOR_MASK_COLOR | PNG_COLOR_MASK_PALETTE)
#define PNG_COLOR_TYPE_RGB        (PNG_COLOR_MASK_COLOR)
#define PNG_COLOR_TYPE_RGB_ALPHA  (PNG_COLOR_MASK_COLOR | PNG_COLOR_MASK_ALPHA)
#define PNG_COLOR_TYPE_GRAY_ALPHA (PNG_COLOR_MASK_ALPHA)
/* aliases */
#define PNG_COLOR_TYPE_RGBA  PNG_COLOR_TYPE_RGB_ALPHA
#define PNG_COLOR_TYPE_GA  PNG_COLOR_TYPE_GRAY_ALPHA

/* This is for compression type. PNG 1.0-1.2 only define the single type. */
```

```
#define PNG_COMPRESSION_TYPE_BASE 0 /* Deflate method 8, 32K window */
#define PNG_COMPRESSION_TYPE_DEFAULT PNG_COMPRESSION_TYPE_BASE

/* This is for filter type. PNG 1.0-1.2 only define the single type. */
#define PNG_FILTER_TYPE_BASE      0 /* Single row per-byte filtering */
#define PNG_INTRAPIXEL_DIFFERENCING 64 /* Used only in MNG datastreams */
#define PNG_FILTER_TYPE_DEFAULT   PNG_FILTER_TYPE_BASE

/* These are for the interlacing type.  These values should NOT be changed. */
#define PNG_INTERLACE_NONE        0 /* Non-interlaced image */
#define PNG_INTERLACE_ADAM7       1 /* Adam7 interlacing */
#define PNG_INTERLACE_LAST        2 /* Not a valid value */

/* These are for the oFFs chunk.  These values should NOT be changed. */
#define PNG_OFFSET_PIXEL          0 /* Offset in pixels */
#define PNG_OFFSET_MICROMETER     1 /* Offset in micrometers (1/10^6 meter) */
#define PNG_OFFSET_LAST           2 /* Not a valid value */

/* These are for the pCAL chunk.  These values should NOT be changed. */
#define PNG_EQUATION_LINEAR       0 /* Linear transformation */
#define PNG_EQUATION_BASE_E       1 /* Exponential base e transform */
#define PNG_EQUATION_ARBITRARY    2 /* Arbitrary base exponential transform */
#define PNG_EQUATION_HYPERBOLIC   3 /* Hyperbolic sine transformation */
#define PNG_EQUATION_LAST         4 /* Not a valid value */

/* These are for the sCAL chunk.  These values should NOT be changed. */
#define PNG_SCALE_UNKNOWN         0 /* unknown unit (image scale) */
#define PNG_SCALE_METER           1 /* meters per pixel */
#define PNG_SCALE_RADIAN          2 /* radians per pixel */
#define PNG_SCALE_LAST            3 /* Not a valid value */

/* These are for the pHYs chunk.  These values should NOT be changed. */
#define PNG_RESOLUTION_UNKNOWN    0 /* pixels/unknown unit (aspect ratio) */
#define PNG_RESOLUTION_METER      1 /* pixels/meter */
#define PNG_RESOLUTION_LAST       2 /* Not a valid value */

/* These are for the sRGB chunk.  These values should NOT be changed. */
#define PNG_sRGB_INTENT_PERCEPTUAL 0
#define PNG_sRGB_INTENT_RELATIVE   1
#define PNG_sRGB_INTENT_SATURATION 2
#define PNG_sRGB_INTENT_ABSOLUTE   3
#define PNG_sRGB_INTENT_LAST       4 /* Not a valid value */

/* This is for text chunks */
#define PNG_KEYWORD_MAX_LENGTH     79

/* Maximum number of entries in PLTE/sPLT/tRNS arrays */
#define PNG_MAX_PALETTE_LENGTH    256

/* These determine if an ancillary chunk's data has been successfully read
 * from the PNG header, or if the application has filled in the corresponding
 * data in the info_struct to be written into the output file.  The values
 * of the PNG_INFO_<chunk> defines should NOT be changed.
 */
```

```
#define PNG_INFO_gAMA 0x0001
#define PNG_INFO_sBIT 0x0002
#define PNG_INFO_cHRM 0x0004
#define PNG_INFO_PLTE 0x0008
#define PNG_INFO_tRNS 0x0010
#define PNG_INFO_bKGD 0x0020
#define PNG_INFO_hIST 0x0040
#define PNG_INFO_pHYs 0x0080
#define PNG_INFO_oFFs 0x0100
#define PNG_INFO_tIME 0x0200
#define PNG_INFO_pCAL 0x0400
#define PNG_INFO_sRGB 0x0800   /* GR-P, 0.96a */
#define PNG_INFO_iCCP 0x1000   /* ESR, 1.0.6 */
#define PNG_INFO_sPLT 0x2000   /* ESR, 1.0.6 */
#define PNG_INFO_sCAL 0x4000   /* ESR, 1.0.6 */
#define PNG_INFO_IDAT 0x8000   /* ESR, 1.0.6 */

/* This is used for the transformation routines, as some of them
 * change these values for the row.  It also should enable using
 * the routines for other purposes.
 */
typedef struct png_row_info_struct
{
   png_uint_32 width;    /* width of row */
   png_size_t rowbytes;  /* number of bytes in row */
   png_byte color_type;  /* color type of row */
   png_byte bit_depth;   /* bit depth of row */
   png_byte channels;    /* number of channels (1, 2, 3, or 4) */
   png_byte pixel_depth; /* bits per pixel (depth * channels) */
} png_row_info;

typedef png_row_info * png_row_infop;
typedef png_row_info * * png_row_infopp;

/* These are the function types for the I/O functions and for the functions
 * that allow the user to override the default I/O functions with his or her
 * own.  The png_error_ptr type should match that of user-supplied warning
 * and error functions, while the png_rw_ptr type should match that of the
 * user read/write data functions.  Note that the 'write' function must not
 * modify the buffer it is passed. The 'read' function, on the other hand, is
 * expected to return the read data in the buffer.
 */
typedef PNG_CALLBACK(void, *png_error_ptr, (png_structp, png_const_charp));
typedef PNG_CALLBACK(void, *png_rw_ptr, (png_structp, png_bytep, png_size_t));
typedef PNG_CALLBACK(void, *png_flush_ptr, (png_structp));
typedef PNG_CALLBACK(void, *png_read_status_ptr, (png_structp, png_uint_32,
   int));
typedef PNG_CALLBACK(void, *png_write_status_ptr, (png_structp, png_uint_32,
   int));

#ifdef PNG_PROGRESSIVE_READ_SUPPORTED
typedef PNG_CALLBACK(void, *png_progressive_info_ptr, (png_structp, png_infop));
typedef PNG_CALLBACK(void, *png_progressive_end_ptr, (png_structp, png_infop));
```

```
/* The following callback receives png_uint_32 row_number, int pass for the
 * png_bytep data of the row.  When transforming an interlaced image the
 * row number is the row number within the sub-image of the interlace pass, so
 * the value will increase to the height of the sub-image (not the full image)
 * then reset to 0 for the next pass.
 *
 * Use PNG_ROW_FROM_PASS_ROW(row, pass) and PNG_COL_FROM_PASS_COL(col, pass) to
 * find the output pixel (x,y) given an interlaced sub-image pixel
 * (row,col,pass).  (See below for these macros.)
 */
typedef PNG_CALLBACK(void, *png_progressive_row_ptr, (png_structp, png_bytep,
   png_uint_32, int));
#endif

#if defined(PNG_READ_USER_TRANSFORM_SUPPORTED) || \
   defined(PNG_WRITE_USER_TRANSFORM_SUPPORTED)
typedef PNG_CALLBACK(void, *png_user_transform_ptr, (png_structp, png_row_infop,
   png_bytep));
#endif

#ifdef PNG_USER_CHUNKS_SUPPORTED
typedef PNG_CALLBACK(int, *png_user_chunk_ptr, (png_structp,
   png_unknown_chunkp));
#endif
#ifdef PNG_UNKNOWN_CHUNKS_SUPPORTED
/* not used anywhere */
/* typedef PNG_CALLBACK(void, *png_unknown_chunk_ptr, (png_structp)); */
#endif

#ifdef PNG_SETJMP_SUPPORTED
/* This must match the function definition in <setjmp.h>, and the application
 * must include this before png.h to obtain the definition of jmp_buf.  The
 * function is required to be PNG_NORETURN, but this is not checked.  If the
 * function does return the application will crash via an abort() or similar
 * system level call.
 *
 * If you get a warning here while building the library you may need to make
 * changes to ensure that pnglibconf.h records the calling convention used by
 * your compiler.  This may be very difficult - try using a different compiler
 * to build the library!
 */
PNG_FUNCTION(void, (PNGCAPI *png_longjmp_ptr), PNGARG((jmp_buf, int)), typedef);
#endif

/* Transform masks for the high-level interface */
#define PNG_TRANSFORM_IDENTITY       0x0000    /* read and write */
#define PNG_TRANSFORM_STRIP_16       0x0001    /* read only */
#define PNG_TRANSFORM_STRIP_ALPHA    0x0002    /* read only */
#define PNG_TRANSFORM_PACKING        0x0004    /* read and write */
#define PNG_TRANSFORM_PACKSWAP       0x0008    /* read and write */
#define PNG_TRANSFORM_EXPAND         0x0010    /* read only */
#define PNG_TRANSFORM_INVERT_MONO    0x0020    /* read and write */
#define PNG_TRANSFORM_SHIFT          0x0040    /* read and write */
#define PNG_TRANSFORM_BGR            0x0080    /* read and write */
```

```
#define PNG_TRANSFORM_SWAP_ALPHA     0x0100    /* read and write */
#define PNG_TRANSFORM_SWAP_ENDIAN    0x0200    /* read and write */
#define PNG_TRANSFORM_INVERT_ALPHA   0x0400    /* read and write */
#define PNG_TRANSFORM_STRIP_FILLER   0x0800    /* write only */
/* Added to libpng-1.2.34 */
#define PNG_TRANSFORM_STRIP_FILLER_BEFORE PNG_TRANSFORM_STRIP_FILLER
#define PNG_TRANSFORM_STRIP_FILLER_AFTER 0x1000 /* write only */
/* Added to libpng-1.4.0 */
#define PNG_TRANSFORM_GRAY_TO_RGB   0x2000      /* read only */
/* Added to libpng-1.5.4 */
#define PNG_TRANSFORM_EXPAND_16     0x4000      /* read only */
#define PNG_TRANSFORM_SCALE_16      0x8000      /* read only */

/* Flags for MNG supported features */
#define PNG_FLAG_MNG_EMPTY_PLTE     0x01
#define PNG_FLAG_MNG_FILTER_64      0x04
#define PNG_ALL_MNG_FEATURES        0x05

/* NOTE: prior to 1.5 these functions had no 'API' style declaration,
 * this allowed the zlib default functions to be used on Windows
 * platforms.  In 1.5 the zlib default malloc (which just calls malloc and
 * ignores the first argument) should be completely compatible with the
 * following.
 */
typedef PNG_CALLBACK(png_voidp, *png_malloc_ptr, (png_structp,
   png_alloc_size_t));
typedef PNG_CALLBACK(void, *png_free_ptr, (png_structp, png_voidp));

/* Section 3: exported functions
 * Here are the function definitions most commonly used.  This is not
 * the place to find out how to use libpng.  See libpng-manual.txt for the
 * full explanation, see example.c for the summary.  This just provides
 * a simple one line description of the use of each function.
 *
 * The PNG_EXPORT() and PNG_EXPORTA() macros used below are defined in
 * pngconf.h and in the *.dfn files in the scripts directory.
 *
 *   PNG_EXPORT(ordinal, type, name, (args));
 *
 *      ordinal:    ordinal that is used while building
 *                  *.def files. The ordinal value is only
 *                  relevant when preprocessing png.h with
 *                  the *.dfn files for building symbol table
 *                  entries, and are removed by pngconf.h.
 *      type:       return type of the function
 *      name:       function name
 *      args:       function arguments, with types
 *
 * When we wish to append attributes to a function prototype we use
 * the PNG_EXPORTA() macro instead.
 *
 *   PNG_EXPORTA(ordinal, type, name, (args), attributes);
 *
 *      ordinal, type, name, and args: same as in PNG_EXPORT().
```

```
 *      attributes: function attributes
 */

/* Returns the version number of the library */
PNG_EXPORT(1, png_uint_32, png_access_version_number, (void));

/* Tell lib we have already handled the first <num_bytes> magic bytes.
 * Handling more than 8 bytes from the beginning of the file is an error.
 */
PNG_EXPORT(2, void, png_set_sig_bytes, (png_structrp png_ptr, int num_bytes));

/* Check sig[start] through sig[start + num_to_check - 1] to see if it's a
 * PNG file.  Returns zero if the supplied bytes match the 8-byte PNG
 * signature, and non-zero otherwise.  Having num_to_check == 0 or
 * start > 7 will always fail (ie return non-zero).
 */
PNG_EXPORT(3, int, png_sig_cmp, (png_const_bytep sig, png_size_t start,
    png_size_t num_to_check));

/* Simple signature checking function.  This is the same as calling
 * png_check_sig(sig, n) := !png_sig_cmp(sig, 0, n).
 */
#define png_check_sig(sig, n) !png_sig_cmp((sig), 0, (n))

/* Allocate and initialize png_ptr struct for reading, and any other memory. */
PNG_EXPORTA(4, png_structp, png_create_read_struct,
    (png_const_charp user_png_ver, png_voidp error_ptr,
    png_error_ptr error_fn, png_error_ptr warn_fn),
    PNG_ALLOCATED);

/* Allocate and initialize png_ptr struct for writing, and any other memory */
PNG_EXPORTA(5, png_structp, png_create_write_struct,
    (png_const_charp user_png_ver, png_voidp error_ptr, png_error_ptr error_fn,
    png_error_ptr warn_fn),
    PNG_ALLOCATED);

PNG_EXPORT(6, png_size_t, png_get_compression_buffer_size,
    (png_const_structrp png_ptr));

PNG_EXPORT(7, void, png_set_compression_buffer_size, (png_structrp png_ptr,
    png_size_t size));

/* Moved from pngconf.h in 1.4.0 and modified to ensure setjmp/longjmp
 * match up.
 */
#ifdef PNG_SETJMP_SUPPORTED
/* This function returns the jmp_buf built in to *png_ptr.  It must be
 * supplied with an appropriate 'longjmp' function to use on that jmp_buf
 * unless the default error function is overridden in which case NULL is
 * acceptable.  The size of the jmp_buf is checked against the actual size
 * allocated by the library - the call will return NULL on a mismatch
 * indicating an ABI mismatch.
 */
PNG_EXPORT(8, jmp_buf*, png_set_longjmp_fn, (png_structrp png_ptr,
```

```
   png_longjmp_ptr longjmp_fn, size_t jmp_buf_size));
#  define png_jmpbuf(png_ptr) \
     (*png_set_longjmp_fn((png_ptr), longjmp, (sizeof (jmp_buf))))
#else
#  define png_jmpbuf(png_ptr) \
     (LIBPNG_WAS_COMPILED_WITH__PNG_NO_SETJMP)
#endif
/* This function should be used by libpng applications in place of
 * longjmp(png_ptr->jmpbuf, val).  If longjmp_fn() has been set, it
 * will use it; otherwise it will call PNG_ABORT().  This function was
 * added in libpng-1.5.0.
 */
PNG_EXPORTA(9, void, png_longjmp, (png_const_structrp png_ptr, int val),
   PNG_NORETURN);

#ifdef PNG_READ_SUPPORTED
/* Reset the compression stream */
PNG_EXPORTA(10, int, png_reset_zstream, (png_structrp png_ptr), PNG_DEPRECATED);
#endif

/* New functions added in libpng-1.0.2 (not enabled by default until 1.2.0) */
#ifdef PNG_USER_MEM_SUPPORTED
PNG_EXPORTA(11, png_structp, png_create_read_struct_2,
   (png_const_charp user_png_ver, png_voidp error_ptr, png_error_ptr error_fn,
   png_error_ptr warn_fn,
   png_voidp mem_ptr, png_malloc_ptr malloc_fn, png_free_ptr free_fn),
   PNG_ALLOCATED);
PNG_EXPORTA(12, png_structp, png_create_write_struct_2,
   (png_const_charp user_png_ver, png_voidp error_ptr, png_error_ptr error_fn,
   png_error_ptr warn_fn,
   png_voidp mem_ptr, png_malloc_ptr malloc_fn, png_free_ptr free_fn),
   PNG_ALLOCATED);
#endif

/* Write the PNG file signature. */
PNG_EXPORT(13, void, png_write_sig, (png_structrp png_ptr));

/* Write a PNG chunk - size, type, (optional) data, CRC. */
PNG_EXPORT(14, void, png_write_chunk, (png_structrp png_ptr, png_const_bytep
   chunk_name, png_const_bytep data, png_size_t length));

/* Write the start of a PNG chunk - length and chunk name. */
PNG_EXPORT(15, void, png_write_chunk_start, (png_structrp png_ptr,
   png_const_bytep chunk_name, png_uint_32 length));

/* Write the data of a PNG chunk started with png_write_chunk_start(). */
PNG_EXPORT(16, void, png_write_chunk_data, (png_structrp png_ptr,
   png_const_bytep data, png_size_t length));

/* Finish a chunk started with png_write_chunk_start() (includes CRC). */
PNG_EXPORT(17, void, png_write_chunk_end, (png_structrp png_ptr));

/* Allocate and initialize the info structure */
PNG_EXPORTA(18, png_infop, png_create_info_struct, (png_const_structrp png_ptr),
```

```
   PNG_ALLOCATED);

/* DEPRECATED: this function allowed init structures to be created using the
 * default allocation method (typically malloc).  Use is deprecated in 1.6.0 and
 * the API will be removed in the future.
 */
PNG_EXPORTA(19, void, png_info_init_3, (png_infopp info_ptr,
   png_size_t png_info_struct_size), PNG_DEPRECATED);

/* Writes all the PNG information before the image. */
PNG_EXPORT(20, void, png_write_info_before_PLTE,
   (png_structrp png_ptr, png_const_inforp info_ptr));
PNG_EXPORT(21, void, png_write_info,
   (png_structrp png_ptr, png_const_inforp info_ptr));

#ifdef PNG_SEQUENTIAL_READ_SUPPORTED
/* Read the information before the actual image data. */
PNG_EXPORT(22, void, png_read_info,
   (png_structrp png_ptr, png_inforp info_ptr));
#endif

#ifdef PNG_TIME_RFC1123_SUPPORTED
   /* Convert to a US string format: there is no localization support in this
    * routine.  The original implementation used a 29 character buffer in
    * png_struct, this will be removed in future versions.
    */
#if PNG_LIBPNG_VER < 10700
/* To do: remove this from libpng17 (and from libpng17/png.c and pngstruct.h) */
PNG_EXPORTA(23, png_const_charp, png_convert_to_rfc1123, (png_structrp png_ptr,
   png_const_timep ptime),PNG_DEPRECATED);
#endif
PNG_EXPORT(241, int, png_convert_to_rfc1123_buffer, (char out[29],
   png_const_timep ptime));
#endif

#ifdef PNG_CONVERT_tIME_SUPPORTED
/* Convert from a struct tm to png_time */
PNG_EXPORT(24, void, png_convert_from_struct_tm, (png_timep ptime,
   const struct tm * ttime));

/* Convert from time_t to png_time.  Uses gmtime() */
PNG_EXPORT(25, void, png_convert_from_time_t, (png_timep ptime, time_t ttime));
#endif /* PNG_CONVERT_tIME_SUPPORTED */

#ifdef PNG_READ_EXPAND_SUPPORTED
/* Expand data to 24-bit RGB, or 8-bit grayscale, with alpha if available. */
PNG_EXPORT(26, void, png_set_expand, (png_structrp png_ptr));
PNG_EXPORT(27, void, png_set_expand_gray_1_2_4_to_8, (png_structrp png_ptr));
PNG_EXPORT(28, void, png_set_palette_to_rgb, (png_structrp png_ptr));
PNG_EXPORT(29, void, png_set_tRNS_to_alpha, (png_structrp png_ptr));
#endif

#ifdef PNG_READ_EXPAND_16_SUPPORTED
/* Expand to 16-bit channels, forces conversion of palette to RGB and expansion
```

```
 * of a tRNS chunk if present.
 */
PNG_EXPORT(221, void, png_set_expand_16, (png_structrp png_ptr));
#endif

#if defined(PNG_READ_BGR_SUPPORTED) || defined(PNG_WRITE_BGR_SUPPORTED)
/* Use blue, green, red order for pixels. */
PNG_EXPORT(30, void, png_set_bgr, (png_structrp png_ptr));
#endif

#ifdef PNG_READ_GRAY_TO_RGB_SUPPORTED
/* Expand the grayscale to 24-bit RGB if necessary. */
PNG_EXPORT(31, void, png_set_gray_to_rgb, (png_structrp png_ptr));
#endif

#ifdef PNG_READ_RGB_TO_GRAY_SUPPORTED
/* Reduce RGB to grayscale. */
#define PNG_ERROR_ACTION_NONE  1
#define PNG_ERROR_ACTION_WARN  2
#define PNG_ERROR_ACTION_ERROR 3
#define PNG_RGB_TO_GRAY_DEFAULT (-1)/*for red/green coefficients*/

PNG_FP_EXPORT(32, void, png_set_rgb_to_gray, (png_structrp png_ptr,
   int error_action, double red, double green))
PNG_FIXED_EXPORT(33, void, png_set_rgb_to_gray_fixed, (png_structrp png_ptr,
   int error_action, png_fixed_point red, png_fixed_point green))

PNG_EXPORT(34, png_byte, png_get_rgb_to_gray_status, (png_const_structrp
   png_ptr));
#endif

#ifdef PNG_BUILD_GRAYSCALE_PALETTE_SUPPORTED
PNG_EXPORT(35, void, png_build_grayscale_palette, (int bit_depth,
   png_colorp palette));
#endif

#ifdef PNG_READ_ALPHA_MODE_SUPPORTED
/* How the alpha channel is interpreted - this affects how the color channels
 * of a PNG file are returned to the calling application when an alpha channel,
 * or a tRNS chunk in a palette file, is present.
 *
 * This has no effect on the way pixels are written into a PNG output
 * datastream. The color samples in a PNG datastream are never premultiplied
 * with the alpha samples.
 *
 * The default is to return data according to the PNG specification: the alpha
 * channel is a linear measure of the contribution of the pixel to the
 * corresponding composited pixel, and the color channels are unassociated
 * (not premultiplied).  The gamma encoded color channels must be scaled
 * according to the contribution and to do this it is necessary to undo
 * the encoding, scale the color values, perform the composition and reencode
 * the values.  This is the 'PNG' mode.
 *
 * The alternative is to 'associate' the alpha with the color information by
```

```
 * storing color channel values that have been scaled by the alpha.
 * image.  These are the 'STANDARD', 'ASSOCIATED' or 'PREMULTIPLIED' modes
 * (the latter being the two common names for associated alpha color channels).
 *
 * For the 'OPTIMIZED' mode, a pixel is treated as opaque only if the alpha
 * value is equal to the maximum value.
 *
 * The final choice is to gamma encode the alpha channel as well.  This is
 * broken because, in practice, no implementation that uses this choice
 * correctly undoes the encoding before handling alpha composition.  Use this
 * choice only if other serious errors in the software or hardware you use
 * mandate it; the typical serious error is for dark halos to appear around
 * opaque areas of the composited PNG image because of arithmetic overflow.
 *
 * The API function png_set_alpha_mode specifies which of these choices to use
 * with an enumerated 'mode' value and the gamma of the required output:
 */
#define PNG_ALPHA_PNG          0 /* according to the PNG standard */
#define PNG_ALPHA_STANDARD      1 /* according to Porter/Duff */
#define PNG_ALPHA_ASSOCIATED    1 /* as above; this is the normal practice */
#define PNG_ALPHA_PREMULTIPLIED 1 /* as above */
#define PNG_ALPHA_OPTIMIZED     2 /* 'PNG' for opaque pixels, else 'STANDARD' */
#define PNG_ALPHA_BROKEN        3 /* the alpha channel is gamma encoded */

PNG_FP_EXPORT(227, void, png_set_alpha_mode, (png_structrp png_ptr, int mode,
   double output_gamma))
PNG_FIXED_EXPORT(228, void, png_set_alpha_mode_fixed, (png_structrp png_ptr,
   int mode, png_fixed_point output_gamma))
#endif

#if defined(PNG_GAMMA_SUPPORTED) || defined(PNG_READ_ALPHA_MODE_SUPPORTED)
/* The output_gamma value is a screen gamma in libpng terminology: it expresses
 * how to decode the output values, not how they are encoded.
 */
#define PNG_DEFAULT_sRGB -1       /* sRGB gamma and color space */
#define PNG_GAMMA_MAC_18 -2       /* Old Mac '1.8' gamma and color space */
#define PNG_GAMMA_sRGB   220000   /* Television standards--matches sRGB gamma */
#define PNG_GAMMA_LINEAR PNG_FP_1 /* Linear */
#endif

/* The following are examples of calls to png_set_alpha_mode to achieve the
 * required overall gamma correction and, where necessary, alpha
 * premultiplication.
 *
 * png_set_alpha_mode(pp, PNG_ALPHA_PNG, PNG_DEFAULT_sRGB);
 *    This is the default libpng handling of the alpha channel - it is not
 *    pre-multiplied into the color components.  In addition the call states
 *    that the output is for a sRGB system and causes all PNG files without gAMA
 *    chunks to be assumed to be encoded using sRGB.
 *
 * png_set_alpha_mode(pp, PNG_ALPHA_PNG, PNG_GAMMA_MAC);
 *    In this case the output is assumed to be something like an sRGB conformant
 *    display preceeded by a power-law lookup table of power 1.45.  This is how
 *    early Mac systems behaved.
```

```
*
* png_set_alpha_mode(pp, PNG_ALPHA_STANDARD, PNG_GAMMA_LINEAR);
*    This is the classic Jim Blinn approach and will work in academic
*    environments where everything is done by the book.  It has the shortcoming
*    of assuming that input PNG data with no gamma information is linear - this
*    is unlikely to be correct unless the PNG files where generated locally.
*    Most of the time the output precision will be so low as to show
*    significant banding in dark areas of the image.
*
* png_set_expand_16(pp);
* png_set_alpha_mode(pp, PNG_ALPHA_STANDARD, PNG_DEFAULT_sRGB);
*    This is a somewhat more realistic Jim Blinn inspired approach.  PNG files
*    are assumed to have the sRGB encoding if not marked with a gamma value and
*    the output is always 16 bits per component.  This permits accurate scaling
*    and processing of the data.  If you know that your input PNG files were
*    generated locally you might need to replace PNG_DEFAULT_sRGB with the
*    correct value for your system.
*
* png_set_alpha_mode(pp, PNG_ALPHA_OPTIMIZED, PNG_DEFAULT_sRGB);
*    If you just need to composite the PNG image onto an existing background
*    and if you control the code that does this you can use the optimization
*    setting.  In this case you just copy completely opaque pixels to the
*    output.  For pixels that are not completely transparent (you just skip
*    those) you do the composition math using png_composite or png_composite_16
*    below then encode the resultant 8-bit or 16-bit values to match the output
*    encoding.
*
* Other cases
*    If neither the PNG nor the standard linear encoding work for you because
*    of the software or hardware you use then you have a big problem.  The PNG
*    case will probably result in halos around the image.  The linear encoding
*    will probably result in a washed out, too bright, image (it's actually too
*    contrasty.)  Try the ALPHA_OPTIMIZED mode above - this will probably
*    substantially reduce the halos.  Alternatively try:
*
* png_set_alpha_mode(pp, PNG_ALPHA_BROKEN, PNG_DEFAULT_sRGB);
*    This option will also reduce the halos, but there will be slight dark
*    halos round the opaque parts of the image where the background is light.
*    In the OPTIMIZED mode the halos will be light halos where the background
*    is dark.  Take your pick - the halos are unavoidable unless you can get
*    your hardware/software fixed!  (The OPTIMIZED approach is slightly
*    faster.)
*
* When the default gamma of PNG files doesn't match the output gamma.
*    If you have PNG files with no gamma information png_set_alpha_mode allows
*    you to provide a default gamma, but it also sets the ouput gamma to the
*    matching value.  If you know your PNG files have a gamma that doesn't
*    match the output you can take advantage of the fact that
*    png_set_alpha_mode always sets the output gamma but only sets the PNG
*    default if it is not already set:
*
* png_set_alpha_mode(pp, PNG_ALPHA_PNG, PNG_DEFAULT_sRGB);
* png_set_alpha_mode(pp, PNG_ALPHA_PNG, PNG_GAMMA_MAC);
*    The first call sets both the default and the output gamma values, the
```

```
 *    second call overrides the output gamma without changing the default.  This
 *    is easier than achieving the same effect with png_set_gamma.  You must use
 *    PNG_ALPHA_PNG for the first call - internal checking in png_set_alpha will
 *    fire if more than one call to png_set_alpha_mode and png_set_background is
 *    made in the same read operation, however multiple calls with PNG_ALPHA_PNG
 *    are ignored.
 */

#ifdef PNG_READ_STRIP_ALPHA_SUPPORTED
PNG_EXPORT(36, void, png_set_strip_alpha, (png_structrp png_ptr));
#endif

#if defined(PNG_READ_SWAP_ALPHA_SUPPORTED) || \
   defined(PNG_WRITE_SWAP_ALPHA_SUPPORTED)
PNG_EXPORT(37, void, png_set_swap_alpha, (png_structrp png_ptr));
#endif

#if defined(PNG_READ_INVERT_ALPHA_SUPPORTED) || \
   defined(PNG_WRITE_INVERT_ALPHA_SUPPORTED)
PNG_EXPORT(38, void, png_set_invert_alpha, (png_structrp png_ptr));
#endif

#if defined(PNG_READ_FILLER_SUPPORTED) || defined(PNG_WRITE_FILLER_SUPPORTED)
/* Add a filler byte to 8-bit Gray or 24-bit RGB images. */
PNG_EXPORT(39, void, png_set_filler, (png_structrp png_ptr, png_uint_32 filler,
   int flags));
/* The values of the PNG_FILLER_ defines should NOT be changed */
#  define PNG_FILLER_BEFORE 0
#  define PNG_FILLER_AFTER 1
/* Add an alpha byte to 8-bit Gray or 24-bit RGB images. */
PNG_EXPORT(40, void, png_set_add_alpha, (png_structrp png_ptr,
   png_uint_32 filler, int flags));
#endif /* PNG_READ_FILLER_SUPPORTED || PNG_WRITE_FILLER_SUPPORTED */

#if defined(PNG_READ_SWAP_SUPPORTED) || defined(PNG_WRITE_SWAP_SUPPORTED)
/* Swap bytes in 16-bit depth files. */
PNG_EXPORT(41, void, png_set_swap, (png_structrp png_ptr));
#endif

#if defined(PNG_READ_PACK_SUPPORTED) || defined(PNG_WRITE_PACK_SUPPORTED)
/* Use 1 byte per pixel in 1, 2, or 4-bit depth files. */
PNG_EXPORT(42, void, png_set_packing, (png_structrp png_ptr));
#endif

#if defined(PNG_READ_PACKSWAP_SUPPORTED) || \
   defined(PNG_WRITE_PACKSWAP_SUPPORTED)
/* Swap packing order of pixels in bytes. */
PNG_EXPORT(43, void, png_set_packswap, (png_structrp png_ptr));
#endif

#if defined(PNG_READ_SHIFT_SUPPORTED) || defined(PNG_WRITE_SHIFT_SUPPORTED)
/* Converts files to legal bit depths. */
PNG_EXPORT(44, void, png_set_shift, (png_structrp png_ptr, png_const_color_8p
   true_bits));
```

```
#endif

#if defined(PNG_READ_INTERLACING_SUPPORTED) || \
    defined(PNG_WRITE_INTERLACING_SUPPORTED)
/* Have the code handle the interlacing.  Returns the number of passes.
 * MUST be called before png_read_update_info or png_start_read_image,
 * otherwise it will not have the desired effect.  Note that it is still
 * necessary to call png_read_row or png_read_rows png_get_image_height
 * times for each pass.
*/
PNG_EXPORT(45, int, png_set_interlace_handling, (png_structrp png_ptr));
#endif

#if defined(PNG_READ_INVERT_SUPPORTED) || defined(PNG_WRITE_INVERT_SUPPORTED)
/* Invert monochrome files */
PNG_EXPORT(46, void, png_set_invert_mono, (png_structrp png_ptr));
#endif

#ifdef PNG_READ_BACKGROUND_SUPPORTED
/* Handle alpha and tRNS by replacing with a background color.  Prior to
 * libpng-1.5.4 this API must not be called before the PNG file header has been
 * read.  Doing so will result in unexpected behavior and possible warnings or
 * errors if the PNG file contains a bKGD chunk.
 */
PNG_FP_EXPORT(47, void, png_set_background, (png_structrp png_ptr,
    png_const_color_16p background_color, int background_gamma_code,
    int need_expand, double background_gamma))
PNG_FIXED_EXPORT(215, void, png_set_background_fixed, (png_structrp png_ptr,
    png_const_color_16p background_color, int background_gamma_code,
    int need_expand, png_fixed_point background_gamma))
#endif
#ifdef PNG_READ_BACKGROUND_SUPPORTED
#  define PNG_BACKGROUND_GAMMA_UNKNOWN 0
#  define PNG_BACKGROUND_GAMMA_SCREEN  1
#  define PNG_BACKGROUND_GAMMA_FILE    2
#  define PNG_BACKGROUND_GAMMA_UNIQUE  3
#endif

#ifdef PNG_READ_SCALE_16_TO_8_SUPPORTED
/* Scale a 16-bit depth file down to 8-bit, accurately. */
PNG_EXPORT(229, void, png_set_scale_16, (png_structrp png_ptr));
#endif

#ifdef PNG_READ_STRIP_16_TO_8_SUPPORTED
#define PNG_READ_16_TO_8 SUPPORTED /* Name prior to 1.5.4 */
/* Strip the second byte of information from a 16-bit depth file. */
PNG_EXPORT(48, void, png_set_strip_16, (png_structrp png_ptr));
#endif

#ifdef PNG_READ_QUANTIZE_SUPPORTED
/* Turn on quantizing, and reduce the palette to the number of colors
 * available.
 */
PNG_EXPORT(49, void, png_set_quantize, (png_structrp png_ptr,
```

```
   png_colorp palette, int num_palette, int maximum_colors,
   png_const_uint_16p histogram, int full_quantize));
#endif

#ifdef PNG_READ_GAMMA_SUPPORTED
/* The threshold on gamma processing is configurable but hard-wired into the
 * library.  The following is the floating point variant.
 */
#define PNG_GAMMA_THRESHOLD (PNG_GAMMA_THRESHOLD_FIXED*.00001)

/* Handle gamma correction. Screen_gamma=(display_exponent).
 * NOTE: this API simply sets the screen and file gamma values. It will
 * therefore override the value for gamma in a PNG file if it is called after
 * the file header has been read - use with care  - call before reading the PNG
 * file for best results!
 *
 * These routines accept the same gamma values as png_set_alpha_mode (described
 * above).  The PNG_GAMMA_ defines and PNG_DEFAULT_sRGB can be passed to either
 * API (floating point or fixed.)  Notice, however, that the 'file_gamma' value
 * is the inverse of a 'screen gamma' value.
 */
PNG_FP_EXPORT(50, void, png_set_gamma, (png_structrp png_ptr,
   double screen_gamma, double override_file_gamma))
PNG_FIXED_EXPORT(208, void, png_set_gamma_fixed, (png_structrp png_ptr,
   png_fixed_point screen_gamma, png_fixed_point override_file_gamma))
#endif

#ifdef PNG_WRITE_FLUSH_SUPPORTED
/* Set how many lines between output flushes - 0 for no flushing */
PNG_EXPORT(51, void, png_set_flush, (png_structrp png_ptr, int nrows));
/* Flush the current PNG output buffer */
PNG_EXPORT(52, void, png_write_flush, (png_structrp png_ptr));
#endif

/* Optional update palette with requested transformations */
PNG_EXPORT(53, void, png_start_read_image, (png_structrp png_ptr));

/* Optional call to update the users info structure */
PNG_EXPORT(54, void, png_read_update_info, (png_structrp png_ptr,
   png_inforp info_ptr));

#ifdef PNG_SEQUENTIAL_READ_SUPPORTED
/* Read one or more rows of image data. */
PNG_EXPORT(55, void, png_read_rows, (png_structrp png_ptr, png_bytepp row,
   png_bytepp display_row, png_uint_32 num_rows));
#endif

#ifdef PNG_SEQUENTIAL_READ_SUPPORTED
/* Read a row of data. */
PNG_EXPORT(56, void, png_read_row, (png_structrp png_ptr, png_bytep row,
   png_bytep display_row));
#endif

#ifdef PNG_SEQUENTIAL_READ_SUPPORTED
```

```
/* Read the whole image into memory at once. */
PNG_EXPORT(57, void, png_read_image, (png_structrp png_ptr, png_bytepp image));
#endif

/* Write a row of image data */
PNG_EXPORT(58, void, png_write_row, (png_structrp png_ptr,
   png_const_bytep row));

/* Write a few rows of image data: (*row) is not written; however, the type
 * is declared as writeable to maintain compatibility with previous versions
 * of libpng and to allow the 'display_row' array from read_rows to be passed
 * unchanged to write_rows.
 */
PNG_EXPORT(59, void, png_write_rows, (png_structrp png_ptr, png_bytepp row,
   png_uint_32 num_rows));

/* Write the image data */
PNG_EXPORT(60, void, png_write_image, (png_structrp png_ptr, png_bytepp image));

/* Write the end of the PNG file. */
PNG_EXPORT(61, void, png_write_end, (png_structrp png_ptr,
   png_inforp info_ptr));

#ifdef PNG_SEQUENTIAL_READ_SUPPORTED
/* Read the end of the PNG file. */
PNG_EXPORT(62, void, png_read_end, (png_structrp png_ptr, png_inforp info_ptr));
#endif

/* Free any memory associated with the png_info_struct */
PNG_EXPORT(63, void, png_destroy_info_struct, (png_const_structrp png_ptr,
   png_infopp info_ptr_ptr));

/* Free any memory associated with the png_struct and the png_info_structs */
PNG_EXPORT(64, void, png_destroy_read_struct, (png_structpp png_ptr_ptr,
   png_infopp info_ptr_ptr, png_infopp end_info_ptr_ptr));

/* Free any memory associated with the png_struct and the png_info_structs */
PNG_EXPORT(65, void, png_destroy_write_struct, (png_structpp png_ptr_ptr,
   png_infopp info_ptr_ptr));

/* Set the libpng method of handling chunk CRC errors */
PNG_EXPORT(66, void, png_set_crc_action, (png_structrp png_ptr, int crit_action,
   int ancil_action));

/* Values for png_set_crc_action() say how to handle CRC errors in
 * ancillary and critical chunks, and whether to use the data contained
 * therein.  Note that it is impossible to "discard" data in a critical
 * chunk.  For versions prior to 0.90, the action was always error/quit,
 * whereas in version 0.90 and later, the action for CRC errors in ancillary
 * chunks is warn/discard.  These values should NOT be changed.
 *
 *      value                  action:critical     action:ancillary
 */
#define PNG_CRC_DEFAULT       0  /* error/quit         warn/discard data */
```

```
#define PNG_CRC_ERROR_QUIT    1  /* error/quit          error/quit       */
#define PNG_CRC_WARN_DISCARD  2  /* (INVALID)           warn/discard data */
#define PNG_CRC_WARN_USE      3  /* warn/use data       warn/use data    */
#define PNG_CRC_QUIET_USE     4  /* quiet/use data      quiet/use data   */
#define PNG_CRC_NO_CHANGE     5  /* use current value   use current value */

/* These functions give the user control over the scan-line filtering in
 * libpng and the compression methods used by zlib.  These functions are
 * mainly useful for testing, as the defaults should work with most users.
 * Those users who are tight on memory or want faster performance at the
 * expense of compression can modify them.  See the compression library
 * header file (zlib.h) for an explination of the compression functions.
 */

/* Set the filtering method(s) used by libpng.  Currently, the only valid
 * value for "method" is 0.
 */
PNG_EXPORT(67, void, png_set_filter, (png_structrp png_ptr, int method,
    int filters));

/* Flags for png_set_filter() to say which filters to use.  The flags
 * are chosen so that they don't conflict with real filter types
 * below, in case they are supplied instead of the #defined constants.
 * These values should NOT be changed.
 */
#define PNG_NO_FILTERS     0x00
#define PNG_FILTER_NONE    0x08
#define PNG_FILTER_SUB     0x10
#define PNG_FILTER_UP      0x20
#define PNG_FILTER_AVG     0x40
#define PNG_FILTER_PAETH   0x80
#define PNG_ALL_FILTERS (PNG_FILTER_NONE | PNG_FILTER_SUB | PNG_FILTER_UP | \
                         PNG_FILTER_AVG | PNG_FILTER_PAETH)

/* Filter values (not flags) - used in pngwrite.c, pngwutil.c for now.
 * These defines should NOT be changed.
 */
#define PNG_FILTER_VALUE_NONE  0
#define PNG_FILTER_VALUE_SUB   1
#define PNG_FILTER_VALUE_UP    2
#define PNG_FILTER_VALUE_AVG   3
#define PNG_FILTER_VALUE_PAETH 4
#define PNG_FILTER_VALUE_LAST  5

#ifdef PNG_WRITE_WEIGHTED_FILTER_SUPPORTED /* EXPERIMENTAL */
/* The "heuristic_method" is given by one of the PNG_FILTER_HEURISTIC_
 * defines, either the default (minimum-sum-of-absolute-differences), or
 * the experimental method (weighted-minimum-sum-of-absolute-differences).
 *
 * Weights are factors >= 1.0, indicating how important it is to keep the
 * filter type consistent between rows.  Larger numbers mean the current
 * filter is that many times as likely to be the same as the "num_weights"
 * previous filters.  This is cumulative for each previous row with a weight.
 * There needs to be "num_weights" values in "filter_weights", or it can be
```

```
 * NULL if the weights aren't being specified.  Weights have no influence on
 * the selection of the first row filter.  Well chosen weights can (in theory)
 * improve the compression for a given image.
 *
 * Costs are factors >= 1.0 indicating the relative decoding costs of a
 * filter type.  Higher costs indicate more decoding expense, and are
 * therefore less likely to be selected over a filter with lower computational
 * costs.  There needs to be a value in "filter_costs" for each valid filter
 * type (given by PNG_FILTER_VALUE_LAST), or it can be NULL if you aren't
 * setting the costs.  Costs try to improve the speed of decompression without
 * unduly increasing the compressed image size.
 *
 * A negative weight or cost indicates the default value is to be used, and
 * values in the range [0.0, 1.0) indicate the value is to remain unchanged.
 * The default values for both weights and costs are currently 1.0, but may
 * change if good general weighting/cost heuristics can be found.  If both
 * the weights and costs are set to 1.0, this degenerates the WEIGHTED method
 * to the UNWEIGHTED method, but with added encoding time/computation.
 */
PNG_FP_EXPORT(68, void, png_set_filter_heuristics, (png_structrp png_ptr,
   int heuristic_method, int num_weights, png_const_doublep filter_weights,
   png_const_doublep filter_costs))
PNG_FIXED_EXPORT(209, void, png_set_filter_heuristics_fixed,
   (png_structrp png_ptr, int heuristic_method, int num_weights,
   png_const_fixed_point_p filter_weights,
   png_const_fixed_point_p filter_costs))
#endif /*  PNG_WRITE_WEIGHTED_FILTER_SUPPORTED */

/* Heuristic used for row filter selection.  These defines should NOT be
 * changed.
 */
#define PNG_FILTER_HEURISTIC_DEFAULT    0  /* Currently "UNWEIGHTED" */
#define PNG_FILTER_HEURISTIC_UNWEIGHTED 1  /* Used by libpng < 0.95 */
#define PNG_FILTER_HEURISTIC_WEIGHTED   2  /* Experimental feature */
#define PNG_FILTER_HEURISTIC_LAST       3  /* Not a valid value */

#ifdef PNG_WRITE_SUPPORTED
/* Set the library compression level.  Currently, valid values range from
 * 0 - 9, corresponding directly to the zlib compression levels 0 - 9
 * (0 - no compression, 9 - "maximal" compression).  Note that tests have
 * shown that zlib compression levels 3-6 usually perform as well as level 9
 * for PNG images, and do considerably fewer caclulations.  In the future,
 * these values may not correspond directly to the zlib compression levels.
 */
PNG_EXPORT(69, void, png_set_compression_level, (png_structrp png_ptr,
   int level));

PNG_EXPORT(70, void, png_set_compression_mem_level, (png_structrp png_ptr,
   int mem_level));

PNG_EXPORT(71, void, png_set_compression_strategy, (png_structrp png_ptr,
   int strategy));

/* If PNG_WRITE_OPTIMIZE_CMF_SUPPORTED is defined, libpng will use a
```

```
 * smaller value of window_bits if it can do so safely.
 */
PNG_EXPORT(72, void, png_set_compression_window_bits, (png_structrp png_ptr,
    int window_bits));

PNG_EXPORT(73, void, png_set_compression_method, (png_structrp png_ptr,
    int method));
#endif

#ifdef PNG_WRITE_CUSTOMIZE_ZTXT_COMPRESSION_SUPPORTED
/* Also set zlib parameters for compressing non-IDAT chunks */
PNG_EXPORT(222, void, png_set_text_compression_level, (png_structrp png_ptr,
    int level));

PNG_EXPORT(223, void, png_set_text_compression_mem_level, (png_structrp png_ptr,
    int mem_level));

PNG_EXPORT(224, void, png_set_text_compression_strategy, (png_structrp png_ptr,
    int strategy));

/* If PNG_WRITE_OPTIMIZE_CMF_SUPPORTED is defined, libpng will use a
 * smaller value of window_bits if it can do so safely.
 */
PNG_EXPORT(225, void, png_set_text_compression_window_bits,
    (png_structrp png_ptr, int window_bits));

PNG_EXPORT(226, void, png_set_text_compression_method, (png_structrp png_ptr,
    int method));
#endif /* PNG_WRITE_CUSTOMIZE_ZTXT_COMPRESSION_SUPPORTED */

/* These next functions are called for input/output, memory, and error
 * handling.  They are in the file pngrio.c, pngwio.c, and pngerror.c,
 * and call standard C I/O routines such as fread(), fwrite(), and
 * fprintf().  These functions can be made to use other I/O routines
 * at run time for those applications that need to handle I/O in a
 * different manner by calling png_set_???_fn().  See libpng-manual.txt for
 * more information.
 */

#ifdef PNG_STDIO_SUPPORTED
/* Initialize the input/output for the PNG file to the default functions. */
PNG_EXPORT(74, void, png_init_io, (png_structrp png_ptr, png_FILE_p fp));
#endif

/* Replace the (error and abort), and warning functions with user
 * supplied functions.  If no messages are to be printed you must still
 * write and use replacement functions. The replacement error_fn should
 * still do a longjmp to the last setjmp location if you are using this
 * method of error handling.  If error_fn or warning_fn is NULL, the
 * default function will be used.
 */

PNG_EXPORT(75, void, png_set_error_fn, (png_structrp png_ptr,
    png_voidp error_ptr, png_error_ptr error_fn, png_error_ptr warning_fn));
```

```
/* Return the user pointer associated with the error functions */
PNG_EXPORT(76, png_voidp, png_get_error_ptr, (png_const_structrp png_ptr));

/* Replace the default data output functions with a user supplied one(s).
 * If buffered output is not used, then output_flush_fn can be set to NULL.
 * If PNG_WRITE_FLUSH_SUPPORTED is not defined at libpng compile time
 * output_flush_fn will be ignored (and thus can be NULL).
 * It is probably a mistake to use NULL for output_flush_fn if
 * write_data_fn is not also NULL unless you have built libpng with
 * PNG_WRITE_FLUSH_SUPPORTED undefined, because in this case libpng's
 * default flush function, which uses the standard *FILE structure, will
 * be used.
 */
PNG_EXPORT(77, void, png_set_write_fn, (png_structrp png_ptr, png_voidp io_ptr,
    png_rw_ptr write_data_fn, png_flush_ptr output_flush_fn));

/* Replace the default data input function with a user supplied one. */
PNG_EXPORT(78, void, png_set_read_fn, (png_structrp png_ptr, png_voidp io_ptr,
    png_rw_ptr read_data_fn));

/* Return the user pointer associated with the I/O functions */
PNG_EXPORT(79, png_voidp, png_get_io_ptr, (png_const_structrp png_ptr));

PNG_EXPORT(80, void, png_set_read_status_fn, (png_structrp png_ptr,
    png_read_status_ptr read_row_fn));

PNG_EXPORT(81, void, png_set_write_status_fn, (png_structrp png_ptr,
    png_write_status_ptr write_row_fn));

#ifdef PNG_USER_MEM_SUPPORTED
/* Replace the default memory allocation functions with user supplied one(s). */
PNG_EXPORT(82, void, png_set_mem_fn, (png_structrp png_ptr, png_voidp mem_ptr,
    png_malloc_ptr malloc_fn, png_free_ptr free_fn));
/* Return the user pointer associated with the memory functions */
PNG_EXPORT(83, png_voidp, png_get_mem_ptr, (png_const_structrp png_ptr));
#endif

#ifdef PNG_READ_USER_TRANSFORM_SUPPORTED
PNG_EXPORT(84, void, png_set_read_user_transform_fn, (png_structrp png_ptr,
    png_user_transform_ptr read_user_transform_fn));
#endif

#ifdef PNG_WRITE_USER_TRANSFORM_SUPPORTED
PNG_EXPORT(85, void, png_set_write_user_transform_fn, (png_structrp png_ptr,
    png_user_transform_ptr write_user_transform_fn));
#endif

#ifdef PNG_USER_TRANSFORM_PTR_SUPPORTED
PNG_EXPORT(86, void, png_set_user_transform_info, (png_structrp png_ptr,
    png_voidp user_transform_ptr, int user_transform_depth,
    int user_transform_channels));
/* Return the user pointer associated with the user transform functions */
PNG_EXPORT(87, png_voidp, png_get_user_transform_ptr,
```

```
   (png_const_structrp png_ptr));
#endif

#ifdef PNG_USER_TRANSFORM_INFO_SUPPORTED
/* Return information about the row currently being processed.  Note that these
 * APIs do not fail but will return unexpected results if called outside a user
 * transform callback.  Also note that when transforming an interlaced image the
 * row number is the row number within the sub-image of the interlace pass, so
 * the value will increase to the height of the sub-image (not the full image)
 * then reset to 0 for the next pass.
 *
 * Use PNG_ROW_FROM_PASS_ROW(row, pass) and PNG_COL_FROM_PASS_COL(col, pass) to
 * find the output pixel (x,y) given an interlaced sub-image pixel
 * (row,col,pass).  (See below for these macros.)
 */
PNG_EXPORT(217, png_uint_32, png_get_current_row_number, (png_const_structrp));
PNG_EXPORT(218, png_byte, png_get_current_pass_number, (png_const_structrp));
#endif

#ifdef PNG_READ_USER_CHUNKS_SUPPORTED
/* This callback is called only for *unknown* chunks.  If
 * PNG_HANDLE_AS_UNKNOWN_SUPPORTED is set then it is possible to set known
 * chunks to be treated as unknown, however in this case the callback must do
 * any processing required by the chunk (e.g. by calling the appropriate
 * png_set_ APIs.)
 *
 * There is no write support - on write, by default, all the chunks in the
 * 'unknown' list are written in the specified position.
 *
 * The integer return from the callback function is interpreted thus:
 *
 * negative: An error occured, png_chunk_error will be called.
 *     zero: The chunk was not handled, the chunk will be saved. A critical
 *           chunk will cause an error at this point unless it is to be saved.
 * positive: The chunk was handled, libpng will ignore/discard it.
 *
 * See "INTERACTION WTIH USER CHUNK CALLBACKS" below for important notes about
 * how this behavior will change in libpng 1.7
 */
PNG_EXPORT(88, void, png_set_read_user_chunk_fn, (png_structrp png_ptr,
   png_voidp user_chunk_ptr, png_user_chunk_ptr read_user_chunk_fn));
#endif

#ifdef PNG_USER_CHUNKS_SUPPORTED
PNG_EXPORT(89, png_voidp, png_get_user_chunk_ptr, (png_const_structrp png_ptr));
#endif

#ifdef PNG_PROGRESSIVE_READ_SUPPORTED
/* Sets the function callbacks for the push reader, and a pointer to a
 * user-defined structure available to the callback functions.
 */
PNG_EXPORT(90, void, png_set_progressive_read_fn, (png_structrp png_ptr,
   png_voidp progressive_ptr, png_progressive_info_ptr info_fn,
   png_progressive_row_ptr row_fn, png_progressive_end_ptr end_fn));
```

```
/* Returns the user pointer associated with the push read functions */
PNG_EXPORT(91, png_voidp, png_get_progressive_ptr,
   (png_const_structrp png_ptr));

/* Function to be called when data becomes available */
PNG_EXPORT(92, void, png_process_data, (png_structrp png_ptr,
   png_inforp info_ptr, png_bytep buffer, png_size_t buffer_size));

/* A function which may be called *only* within png_process_data to stop the
 * processing of any more data.  The function returns the number of bytes
 * remaining, excluding any that libpng has cached internally.  A subsequent
 * call to png_process_data must supply these bytes again.  If the argument
 * 'save' is set to true the routine will first save all the pending data and
 * will always return 0.
 */
PNG_EXPORT(219, png_size_t, png_process_data_pause, (png_structrp, int save));

/* A function which may be called *only* outside (after) a call to
 * png_process_data.  It returns the number of bytes of data to skip in the
 * input.  Normally it will return 0, but if it returns a non-zero value the
 * application must skip than number of bytes of input data and pass the
 * following data to the next call to png_process_data.
 */
PNG_EXPORT(220, png_uint_32, png_process_data_skip, (png_structrp));

/* Function that combines rows.  'new_row' is a flag that should come from
 * the callback and be non-NULL if anything needs to be done; the library
 * stores its own version of the new data internally and ignores the passed
 * in value.
 */
PNG_EXPORT(93, void, png_progressive_combine_row, (png_const_structrp png_ptr,
   png_bytep old_row, png_const_bytep new_row));
#endif /* PNG_PROGRESSIVE_READ_SUPPORTED */

PNG_EXPORTA(94, png_voidp, png_malloc, (png_const_structrp png_ptr,
   png_alloc_size_t size), PNG_ALLOCATED);
/* Added at libpng version 1.4.0 */
PNG_EXPORTA(95, png_voidp, png_calloc, (png_const_structrp png_ptr,
   png_alloc_size_t size), PNG_ALLOCATED);

/* Added at libpng version 1.2.4 */
PNG_EXPORTA(96, png_voidp, png_malloc_warn, (png_const_structrp png_ptr,
   png_alloc_size_t size), PNG_ALLOCATED);

/* Frees a pointer allocated by png_malloc() */
PNG_EXPORT(97, void, png_free, (png_const_structrp png_ptr, png_voidp ptr));

/* Free data that was allocated internally */
PNG_EXPORT(98, void, png_free_data, (png_const_structrp png_ptr,
   png_inforp info_ptr, png_uint_32 free_me, int num));

/* Reassign responsibility for freeing existing data, whether allocated
 * by libpng or by the application; this works on the png_info structure passed
```

```
 * in, it does not change the state for other png_info structures.
 *
 * It is unlikely that this function works correctly as of 1.6.0 and using it
 * may result either in memory leaks or double free of allocated data.
 */
PNG_EXPORT(99, void, png_data_freer, (png_const_structrp png_ptr,
    png_inforp info_ptr, int freer, png_uint_32 mask));

/* Assignments for png_data_freer */
#define PNG_DESTROY_WILL_FREE_DATA 1
#define PNG_SET_WILL_FREE_DATA 1
#define PNG_USER_WILL_FREE_DATA 2
/* Flags for png_ptr->free_me and info_ptr->free_me */
#define PNG_FREE_HIST 0x0008
#define PNG_FREE_ICCP 0x0010
#define PNG_FREE_SPLT 0x0020
#define PNG_FREE_ROWS 0x0040
#define PNG_FREE_PCAL 0x0080
#define PNG_FREE_SCAL 0x0100
#ifdef PNG_STORE_UNKNOWN_CHUNKS_SUPPORTED
#  define PNG_FREE_UNKN 0x0200
#endif
/*      PNG_FREE_LIST 0x0400    removed in 1.6.0 because it is ignored */
#define PNG_FREE_PLTE 0x1000
#define PNG_FREE_TRNS 0x2000
#define PNG_FREE_TEXT 0x4000
#define PNG_FREE_ALL  0x7fff
#define PNG_FREE_MUL  0x4220 /* PNG_FREE_SPLT|PNG_FREE_TEXT|PNG_FREE_UNKN */

#ifdef PNG_USER_MEM_SUPPORTED
PNG_EXPORTA(100, png_voidp, png_malloc_default, (png_const_structrp png_ptr,
    png_alloc_size_t size), PNG_ALLOCATED PNG_DEPRECATED);
PNG_EXPORTA(101, void, png_free_default, (png_const_structrp png_ptr,
    png_voidp ptr), PNG_DEPRECATED);
#endif

#ifdef PNG_ERROR_TEXT_SUPPORTED
/* Fatal error in PNG image of libpng - can't continue */
PNG_EXPORTA(102, void, png_error, (png_const_structrp png_ptr,
    png_const_charp error_message), PNG_NORETURN);

/* The same, but the chunk name is prepended to the error string. */
PNG_EXPORTA(103, void, png_chunk_error, (png_const_structrp png_ptr,
    png_const_charp error_message), PNG_NORETURN);

#else
/* Fatal error in PNG image of libpng - can't continue */
PNG_EXPORTA(104, void, png_err, (png_const_structrp png_ptr), PNG_NORETURN);
#  define png_error(s1,s2) png_err(s1)
#  define png_chunk_error(s1,s2) png_err(s1)
#endif

#ifdef PNG_WARNINGS_SUPPORTED
/* Non-fatal error in libpng.  Can continue, but may have a problem. */
```

```
PNG_EXPORT(105, void, png_warning, (png_const_structrp png_ptr,
   png_const_charp warning_message));

/* Non-fatal error in libpng, chunk name is prepended to message. */
PNG_EXPORT(106, void, png_chunk_warning, (png_const_structrp png_ptr,
   png_const_charp warning_message));
#else
#  define png_warning(s1,s2) ((void)(s1))
#  define png_chunk_warning(s1,s2) ((void)(s1))
#endif

#ifdef PNG_BENIGN_ERRORS_SUPPORTED
/* Benign error in libpng.  Can continue, but may have a problem.
 * User can choose whether to handle as a fatal error or as a warning. */
PNG_EXPORT(107, void, png_benign_error, (png_const_structrp png_ptr,
   png_const_charp warning_message));

#ifdef PNG_READ_SUPPORTED
/* Same, chunk name is prepended to message (only during read) */
PNG_EXPORT(108, void, png_chunk_benign_error, (png_const_structrp png_ptr,
   png_const_charp warning_message));
#endif

PNG_EXPORT(109, void, png_set_benign_errors,
   (png_structrp png_ptr, int allowed));
#else
#  ifdef PNG_ALLOW_BENIGN_ERRORS
#    define png_benign_error png_warning
#    define png_chunk_benign_error png_chunk_warning
#  else
#    define png_benign_error png_error
#    define png_chunk_benign_error png_chunk_error
#  endif
#endif

/* The png_set_<chunk> functions are for storing values in the png_info_struct.
 * Similarly, the png_get_<chunk> calls are used to read values from the
 * png_info_struct, either storing the parameters in the passed variables, or
 * setting pointers into the png_info_struct where the data is stored.  The
 * png_get_<chunk> functions return a non-zero value if the data was available
 * in info_ptr, or return zero and do not change any of the parameters if the
 * data was not available.
 *
 * These functions should be used instead of directly accessing png_info
 * to avoid problems with future changes in the size and internal layout of
 * png_info_struct.
 */
/* Returns "flag" if chunk data is valid in info_ptr. */
PNG_EXPORT(110, png_uint_32, png_get_valid, (png_const_structrp png_ptr,
   png_const_inforp info_ptr, png_uint_32 flag));

/* Returns number of bytes needed to hold a transformed row. */
PNG_EXPORT(111, png_size_t, png_get_rowbytes, (png_const_structrp png_ptr,
   png_const_inforp info_ptr));
```

```
#ifdef PNG_INFO_IMAGE_SUPPORTED
/* Returns row_pointers, which is an array of pointers to scanlines that was
 * returned from png_read_png().
 */
PNG_EXPORT(112, png_bytepp, png_get_rows, (png_const_structrp png_ptr,
    png_const_inforp info_ptr));

/* Set row_pointers, which is an array of pointers to scanlines for use
 * by png_write_png().
 */
PNG_EXPORT(113, void, png_set_rows, (png_const_structrp png_ptr,
    png_inforp info_ptr, png_bytepp row_pointers));
#endif

/* Returns number of color channels in image. */
PNG_EXPORT(114, png_byte, png_get_channels, (png_const_structrp png_ptr,
    png_const_inforp info_ptr));

#ifdef PNG_EASY_ACCESS_SUPPORTED
/* Returns image width in pixels. */
PNG_EXPORT(115, png_uint_32, png_get_image_width, (png_const_structrp png_ptr,
    png_const_inforp info_ptr));

/* Returns image height in pixels. */
PNG_EXPORT(116, png_uint_32, png_get_image_height, (png_const_structrp png_ptr,
    png_const_inforp info_ptr));

/* Returns image bit_depth. */
PNG_EXPORT(117, png_byte, png_get_bit_depth, (png_const_structrp png_ptr,
    png_const_inforp info_ptr));

/* Returns image color_type. */
PNG_EXPORT(118, png_byte, png_get_color_type, (png_const_structrp png_ptr,
    png_const_inforp info_ptr));

/* Returns image filter_type. */
PNG_EXPORT(119, png_byte, png_get_filter_type, (png_const_structrp png_ptr,
    png_const_inforp info_ptr));

/* Returns image interlace_type. */
PNG_EXPORT(120, png_byte, png_get_interlace_type, (png_const_structrp png_ptr,
    png_const_inforp info_ptr));

/* Returns image compression_type. */
PNG_EXPORT(121, png_byte, png_get_compression_type, (png_const_structrp png_ptr,
    png_const_inforp info_ptr));

/* Returns image resolution in pixels per meter, from pHYs chunk data. */
PNG_EXPORT(122, png_uint_32, png_get_pixels_per_meter,
    (png_const_structrp png_ptr, png_const_inforp info_ptr));
PNG_EXPORT(123, png_uint_32, png_get_x_pixels_per_meter,
    (png_const_structrp png_ptr, png_const_inforp info_ptr));
PNG_EXPORT(124, png_uint_32, png_get_y_pixels_per_meter,
```

```
   (png_const_structrp png_ptr, png_const_inforp info_ptr));

/* Returns pixel aspect ratio, computed from pHYs chunk data.  */
PNG_FP_EXPORT(125, float, png_get_pixel_aspect_ratio,
   (png_const_structrp png_ptr, png_const_inforp info_ptr))
PNG_FIXED_EXPORT(210, png_fixed_point, png_get_pixel_aspect_ratio_fixed,
   (png_const_structrp png_ptr, png_const_inforp info_ptr))

/* Returns image x, y offset in pixels or microns, from oFFs chunk data. */
PNG_EXPORT(126, png_int_32, png_get_x_offset_pixels,
   (png_const_structrp png_ptr, png_const_inforp info_ptr));
PNG_EXPORT(127, png_int_32, png_get_y_offset_pixels,
   (png_const_structrp png_ptr, png_const_inforp info_ptr));
PNG_EXPORT(128, png_int_32, png_get_x_offset_microns,
   (png_const_structrp png_ptr, png_const_inforp info_ptr));
PNG_EXPORT(129, png_int_32, png_get_y_offset_microns,
   (png_const_structrp png_ptr, png_const_inforp info_ptr));

#endif /* PNG_EASY_ACCESS_SUPPORTED */

#ifdef PNG_READ_SUPPORTED
/* Returns pointer to signature string read from PNG header */
PNG_EXPORT(130, png_const_bytep, png_get_signature, (png_const_structrp png_ptr,
   png_const_inforp info_ptr));
#endif

#ifdef PNG_bKGD_SUPPORTED
PNG_EXPORT(131, png_uint_32, png_get_bKGD, (png_const_structrp png_ptr,
   png_inforp info_ptr, png_color_16p *background));
#endif

#ifdef PNG_bKGD_SUPPORTED
PNG_EXPORT(132, void, png_set_bKGD, (png_const_structrp png_ptr,
   png_inforp info_ptr, png_const_color_16p background));
#endif

#ifdef PNG_cHRM_SUPPORTED
PNG_FP_EXPORT(133, png_uint_32, png_get_cHRM, (png_const_structrp png_ptr,
   png_const_inforp info_ptr, double *white_x, double *white_y, double *red_x,
   double *red_y, double *green_x, double *green_y, double *blue_x,
   double *blue_y))
PNG_FP_EXPORT(230, png_uint_32, png_get_cHRM_XYZ, (png_const_structrp png_ptr,
   png_const_inforp info_ptr, double *red_X, double *red_Y, double *red_Z,
   double *green_X, double *green_Y, double *green_Z, double *blue_X,
   double *blue_Y, double *blue_Z))
PNG_FIXED_EXPORT(134, png_uint_32, png_get_cHRM_fixed,
   (png_const_structrp png_ptr, png_const_inforp info_ptr,
   png_fixed_point *int_white_x, png_fixed_point *int_white_y,
   png_fixed_point *int_red_x, png_fixed_point *int_red_y,
   png_fixed_point *int_green_x, png_fixed_point *int_green_y,
   png_fixed_point *int_blue_x, png_fixed_point *int_blue_y))
PNG_FIXED_EXPORT(231, png_uint_32, png_get_cHRM_XYZ_fixed,
   (png_const_structrp png_ptr, png_const_inforp info_ptr,
   png_fixed_point *int_red_X, png_fixed_point *int_red_Y,
```

```
    png_fixed_point *int_red_Z, png_fixed_point *int_green_X,
    png_fixed_point *int_green_Y, png_fixed_point *int_green_Z,
    png_fixed_point *int_blue_X, png_fixed_point *int_blue_Y,
    png_fixed_point *int_blue_Z))
#endif

#ifdef PNG_cHRM_SUPPORTED
PNG_FP_EXPORT(135, void, png_set_cHRM, (png_const_structrp png_ptr,
    png_inforp info_ptr,
    double white_x, double white_y, double red_x, double red_y, double green_x,
    double green_y, double blue_x, double blue_y))
PNG_FP_EXPORT(232, void, png_set_cHRM_XYZ, (png_const_structrp png_ptr,
    png_inforp info_ptr, double red_X, double red_Y, double red_Z,
    double green_X, double green_Y, double green_Z, double blue_X,
    double blue_Y, double blue_Z))
PNG_FIXED_EXPORT(136, void, png_set_cHRM_fixed, (png_const_structrp png_ptr,
    png_inforp info_ptr, png_fixed_point int_white_x,
    png_fixed_point int_white_y, png_fixed_point int_red_x,
    png_fixed_point int_red_y, png_fixed_point int_green_x,
    png_fixed_point int_green_y, png_fixed_point int_blue_x,
    png_fixed_point int_blue_y))
PNG_FIXED_EXPORT(233, void, png_set_cHRM_XYZ_fixed, (png_const_structrp png_ptr,
    png_inforp info_ptr, png_fixed_point int_red_X, png_fixed_point int_red_Y,
    png_fixed_point int_red_Z, png_fixed_point int_green_X,
    png_fixed_point int_green_Y, png_fixed_point int_green_Z,
    png_fixed_point int_blue_X, png_fixed_point int_blue_Y,
    png_fixed_point int_blue_Z))
#endif

#ifdef PNG_gAMA_SUPPORTED
PNG_FP_EXPORT(137, png_uint_32, png_get_gAMA, (png_const_structrp png_ptr,
    png_const_inforp info_ptr, double *file_gamma))
PNG_FIXED_EXPORT(138, png_uint_32, png_get_gAMA_fixed,
    (png_const_structrp png_ptr, png_const_inforp info_ptr,
    png_fixed_point *int_file_gamma))
#endif

#ifdef PNG_gAMA_SUPPORTED
PNG_FP_EXPORT(139, void, png_set_gAMA, (png_const_structrp png_ptr,
    png_inforp info_ptr, double file_gamma))
PNG_FIXED_EXPORT(140, void, png_set_gAMA_fixed, (png_const_structrp png_ptr,
    png_inforp info_ptr, png_fixed_point int_file_gamma))
#endif

#ifdef PNG_hIST_SUPPORTED
PNG_EXPORT(141, png_uint_32, png_get_hIST, (png_const_structrp png_ptr,
    png_inforp info_ptr, png_uint_16p *hist));
#endif

#ifdef PNG_hIST_SUPPORTED
PNG_EXPORT(142, void, png_set_hIST, (png_const_structrp png_ptr,
    png_inforp info_ptr, png_const_uint_16p hist));
#endif
```

```
PNG_EXPORT(143, png_uint_32, png_get_IHDR, (png_const_structrp png_ptr,
    png_const_inforp info_ptr, png_uint_32 *width, png_uint_32 *height,
    int *bit_depth, int *color_type, int *interlace_method,
    int *compression_method, int *filter_method));

PNG_EXPORT(144, void, png_set_IHDR, (png_const_structrp png_ptr,
    png_inforp info_ptr, png_uint_32 width, png_uint_32 height, int bit_depth,
    int color_type, int interlace_method, int compression_method,
    int filter_method));

#ifdef PNG_oFFs_SUPPORTED
PNG_EXPORT(145, png_uint_32, png_get_oFFs, (png_const_structrp png_ptr,
   png_const_inforp info_ptr, png_int_32 *offset_x, png_int_32 *offset_y,
   int *unit_type));
#endif

#ifdef PNG_oFFs_SUPPORTED
PNG_EXPORT(146, void, png_set_oFFs, (png_const_structrp png_ptr,
    png_inforp info_ptr, png_int_32 offset_x, png_int_32 offset_y,
    int unit_type));
#endif

#ifdef PNG_pCAL_SUPPORTED
PNG_EXPORT(147, png_uint_32, png_get_pCAL, (png_const_structrp png_ptr,
    png_inforp info_ptr, png_charp *purpose, png_int_32 *X0,
    png_int_32 *X1, int *type, int *nparams, png_charp *units,
    png_charpp *params));
#endif

#ifdef PNG_pCAL_SUPPORTED
PNG_EXPORT(148, void, png_set_pCAL, (png_const_structrp png_ptr,
    png_inforp info_ptr, png_const_charp purpose, png_int_32 X0, png_int_32 X1,
    int type, int nparams, png_const_charp units, png_charpp params));
#endif

#ifdef PNG_pHYs_SUPPORTED
PNG_EXPORT(149, png_uint_32, png_get_pHYs, (png_const_structrp png_ptr,
    png_const_inforp info_ptr, png_uint_32 *res_x, png_uint_32 *res_y,
    int *unit_type));
#endif

#ifdef PNG_pHYs_SUPPORTED
PNG_EXPORT(150, void, png_set_pHYs, (png_const_structrp png_ptr,
    png_inforp info_ptr, png_uint_32 res_x, png_uint_32 res_y, int unit_type));
#endif

PNG_EXPORT(151, png_uint_32, png_get_PLTE, (png_const_structrp png_ptr,
   png_inforp info_ptr, png_colorp *palette, int *num_palette));

PNG_EXPORT(152, void, png_set_PLTE, (png_structrp png_ptr,
    png_inforp info_ptr, png_const_colorp palette, int num_palette));

#ifdef PNG_sBIT_SUPPORTED
PNG_EXPORT(153, png_uint_32, png_get_sBIT, (png_const_structrp png_ptr,
```

```
   png_inforp info_ptr, png_color_8p *sig_bit));
#endif

#ifdef PNG_sBIT_SUPPORTED
PNG_EXPORT(154, void, png_set_sBIT, (png_const_structrp png_ptr,
   png_inforp info_ptr, png_const_color_8p sig_bit));
#endif

#ifdef PNG_sRGB_SUPPORTED
PNG_EXPORT(155, png_uint_32, png_get_sRGB, (png_const_structrp png_ptr,
   png_const_inforp info_ptr, int *file_srgb_intent));
#endif

#ifdef PNG_sRGB_SUPPORTED
PNG_EXPORT(156, void, png_set_sRGB, (png_const_structrp png_ptr,
   png_inforp info_ptr, int srgb_intent));
PNG_EXPORT(157, void, png_set_sRGB_gAMA_and_cHRM, (png_const_structrp png_ptr,
   png_inforp info_ptr, int srgb_intent));
#endif

#ifdef PNG_iCCP_SUPPORTED
PNG_EXPORT(158, png_uint_32, png_get_iCCP, (png_const_structrp png_ptr,
   png_inforp info_ptr, png_charpp name, int *compression_type,
   png_bytepp profile, png_uint_32 *proflen));
#endif

#ifdef PNG_iCCP_SUPPORTED
PNG_EXPORT(159, void, png_set_iCCP, (png_const_structrp png_ptr,
   png_inforp info_ptr, png_const_charp name, int compression_type,
   png_const_bytep profile, png_uint_32 proflen));
#endif

#ifdef PNG_sPLT_SUPPORTED
PNG_EXPORT(160, int, png_get_sPLT, (png_const_structrp png_ptr,
   png_inforp info_ptr, png_sPLT_tpp entries));
#endif

#ifdef PNG_sPLT_SUPPORTED
PNG_EXPORT(161, void, png_set_sPLT, (png_const_structrp png_ptr,
   png_inforp info_ptr, png_const_sPLT_tp entries, int nentries));
#endif

#ifdef PNG_TEXT_SUPPORTED
/* png_get_text also returns the number of text chunks in *num_text */
PNG_EXPORT(162, int, png_get_text, (png_const_structrp png_ptr,
   png_inforp info_ptr, png_textp *text_ptr, int *num_text));
#endif

/* Note while png_set_text() will accept a structure whose text,
 * language, and  translated keywords are NULL pointers, the structure
 * returned by png_get_text will always contain regular
 * zero-terminated C strings.  They might be empty strings but
 * they will never be NULL pointers.
 */
```

```
#ifdef PNG_TEXT_SUPPORTED
PNG_EXPORT(163, void, png_set_text, (png_const_structrp png_ptr,
    png_inforp info_ptr, png_const_textp text_ptr, int num_text));
#endif

#ifdef PNG_tIME_SUPPORTED
PNG_EXPORT(164, png_uint_32, png_get_tIME, (png_const_structrp png_ptr,
    png_inforp info_ptr, png_timep *mod_time));
#endif

#ifdef PNG_tIME_SUPPORTED
PNG_EXPORT(165, void, png_set_tIME, (png_const_structrp png_ptr,
    png_inforp info_ptr, png_const_timep mod_time));
#endif

#ifdef PNG_tRNS_SUPPORTED
PNG_EXPORT(166, png_uint_32, png_get_tRNS, (png_const_structrp png_ptr,
    png_inforp info_ptr, png_bytep *trans_alpha, int *num_trans,
    png_color_16p *trans_color));
#endif

#ifdef PNG_tRNS_SUPPORTED
PNG_EXPORT(167, void, png_set_tRNS, (png_structrp png_ptr,
    png_inforp info_ptr, png_const_bytep trans_alpha, int num_trans,
    png_const_color_16p trans_color));
#endif

#ifdef PNG_sCAL_SUPPORTED
PNG_FP_EXPORT(168, png_uint_32, png_get_sCAL, (png_const_structrp png_ptr,
    png_const_inforp info_ptr, int *unit, double *width, double *height))
#if defined(PNG_FLOATING_ARITHMETIC_SUPPORTED) || \
   defined(PNG_FLOATING_POINT_SUPPORTED)
/* NOTE: this API is currently implemented using floating point arithmetic,
 * consequently it can only be used on systems with floating point support.
 * In any case the range of values supported by png_fixed_point is small and it
 * is highly recommended that png_get_sCAL_s be used instead.
 */
PNG_FIXED_EXPORT(214, png_uint_32, png_get_sCAL_fixed,
    (png_const_structrp png_ptr, png_const_inforp info_ptr, int *unit,
    png_fixed_point *width, png_fixed_point *height))
#endif
PNG_EXPORT(169, png_uint_32, png_get_sCAL_s,
    (png_const_structrp png_ptr, png_const_inforp info_ptr, int *unit,
    png_charpp swidth, png_charpp sheight));

PNG_FP_EXPORT(170, void, png_set_sCAL, (png_const_structrp png_ptr,
    png_inforp info_ptr, int unit, double width, double height))
PNG_FIXED_EXPORT(213, void, png_set_sCAL_fixed, (png_const_structrp png_ptr,
   png_inforp info_ptr, int unit, png_fixed_point width,
   png_fixed_point height))
PNG_EXPORT(171, void, png_set_sCAL_s, (png_const_structrp png_ptr,
    png_inforp info_ptr, int unit,
    png_const_charp swidth, png_const_charp sheight));
```

```
#endif /* PNG_sCAL_SUPPORTED */

#ifdef PNG_SET_UNKNOWN_CHUNKS_SUPPORTED
/* Provide the default handling for all unknown chunks or, optionally, for
 * specific unknown chunks.
 *
 * NOTE: prior to 1.6.0 the handling specified for particular chunks on read was
 * ignored and the default was used, the per-chunk setting only had an effect on
 * write.  If you wish to have chunk-specific handling on read in code that must
 * work on earlier versions you must use a user chunk callback to specify the
 * desired handling (keep or discard.)
 *
 * The 'keep' parameter is a PNG_HANDLE_CHUNK_ value as listed below.  The
 * parameter is interpreted as follows:
 *
 * READ:
 *    PNG_HANDLE_CHUNK_AS_DEFAULT:
 *      Known chunks: do normal libpng processing, do not keep the chunk (but
 *        see the comments below about PNG_HANDLE_AS_UNKNOWN_SUPPORTED)
 *      Unknown chunks: for a specific chunk use the global default, when used
 *        as the default discard the chunk data.
 *    PNG_HANDLE_CHUNK_NEVER:
 *      Discard the chunk data.
 *    PNG_HANDLE_CHUNK_IF_SAFE:
 *      Keep the chunk data if the chunk is not critical else raise a chunk
 *      error.
 *    PNG_HANDLE_CHUNK_ALWAYS:
 *      Keep the chunk data.
 *
 * If the chunk data is saved it can be retrieved using png_get_unknown_chunks,
 * below.  Notice that specifying "AS_DEFAULT" as a global default is equivalent
 * to specifying "NEVER", however when "AS_DEFAULT" is used for specific chunks
 * it simply resets the behavior to the libpng default.
 *
 * INTERACTION WTIH USER CHUNK CALLBACKS:
 * The per-chunk handling is always used when there is a png_user_chunk_ptr
 * callback and the callback returns 0; the chunk is then always stored *unless*
 * it is critical and the per-chunk setting is other than ALWAYS.  Notice that
 * the global default is *not* used in this case.  (In effect the per-chunk
 * value is incremented to at least IF_SAFE.)
 *
 * IMPORTANT NOTE: this behavior will change in libpng 1.7 - the global and
 * per-chunk defaults will be honored.  If you want to preserve the current
 * behavior when your callback returns 0 you must set PNG_HANDLE_CHUNK_IF_SAFE
 * as the default - if you don't do this libpng 1.6 will issue a warning.
 *
 * If you want unhandled unknown chunks to be discarded in libpng 1.6 and
 * earlier simply return '1' (handled).
 *
 * PNG_HANDLE_AS_UNKNOWN_SUPPORTED:
 *    If this is *not* set known chunks will always be handled by libpng and
 *    will never be stored in the unknown chunk list.  Known chunks listed to
 *    png_set_keep_unknown_chunks will have no effect.  If it is set then known
 *    chunks listed with a keep other than AS_DEFAULT will *never* be processed
```

```
 *    by libpng, in addition critical chunks must either be processed by the
 *    callback or saved.
 *
 *    The IHDR and IEND chunks must not be listed.  Because this turns off the
 *    default handling for chunks that would otherwise be recognized the
 *    behavior of libpng transformations may well become incorrect!
 *
 * WRITE:
 *    When writing chunks the options only apply to the chunks specified by
 *    png_set_unknown_chunks (below), libpng will *always* write known chunks
 *    required by png_set_ calls and will always write the core critical chunks
 *    (as required for PLTE).
 *
 *    Each chunk in the png_set_unknown_chunks list is looked up in the
 *    png_set_keep_unknown_chunks list to find the keep setting, this is then
 *    interpreted as follows:
 *
 *    PNG_HANDLE_CHUNK_AS_DEFAULT:
 *      Write safe-to-copy chunks and write other chunks if the global
 *      default is set to _ALWAYS, otherwise don't write this chunk.
 *    PNG_HANDLE_CHUNK_NEVER:
 *      Do not write the chunk.
 *    PNG_HANDLE_CHUNK_IF_SAFE:
 *      Write the chunk if it is safe-to-copy, otherwise do not write it.
 *    PNG_HANDLE_CHUNK_ALWAYS:
 *      Write the chunk.
 *
 * Note that the default behavior is effectively the opposite of the read case -
 * in read unknown chunks are not stored by default, in write they are written
 * by default.  Also the behavior of PNG_HANDLE_CHUNK_IF_SAFE is very different
 * - on write the safe-to-copy bit is checked, on read the critical bit is
 * checked and on read if the chunk is critical an error will be raised.
 *
 * num_chunks:
 * ===========
 *    If num_chunks is positive, then the "keep" parameter specifies the manner
 *    for handling only those chunks appearing in the chunk_list array,
 *    otherwise the chunk list array is ignored.
 *
 *    If num_chunks is 0 the "keep" parameter specifies the default behavior for
 *    unknown chunks, as described above.
 *
 *    If num_chunks is negative, then the "keep" parameter specifies the manner
 *    for handling all unknown chunks plus all chunks recognized by libpng
 *    except for the IHDR, PLTE, tRNS, IDAT, and IEND chunks (which continue to
 *    be processed by libpng.
 */
PNG_EXPORT(172, void, png_set_keep_unknown_chunks, (png_structrp png_ptr,
    int keep, png_const_bytep chunk_list, int num_chunks));

/* The "keep" PNG_HANDLE_CHUNK_ parameter for the specified chunk is returned;
 * the result is therefore true (non-zero) if special handling is required,
 * false for the default handling.
 */
```

```
PNG_EXPORT(173, int, png_handle_as_unknown, (png_const_structrp png_ptr,
    png_const_bytep chunk_name));
#endif

#ifdef PNG_STORE_UNKNOWN_CHUNKS_SUPPORTED
PNG_EXPORT(174, void, png_set_unknown_chunks, (png_const_structrp png_ptr,
    png_inforp info_ptr, png_const_unknown_chunkp unknowns,
    int num_unknowns));
   /* NOTE: prior to 1.6.0 this routine set the 'location' field of the added
    * unknowns to the location currently stored in the png_struct.  This is
    * invariably the wrong value on write.  To fix this call the following API
    * for each chunk in the list with the correct location.  If you know your
    * code won't be compiled on earlier versions you can rely on
    * png_set_unknown_chunks(write-ptr, png_get_unknown_chunks(read-ptr)) doing
    * the correct thing.
    */

PNG_EXPORT(175, void, png_set_unknown_chunk_location,
    (png_const_structrp png_ptr, png_inforp info_ptr, int chunk, int location));

PNG_EXPORT(176, int, png_get_unknown_chunks, (png_const_structrp png_ptr,
    png_inforp info_ptr, png_unknown_chunkpp entries));
#endif

/* Png_free_data() will turn off the "valid" flag for anything it frees.
 * If you need to turn it off for a chunk that your application has freed,
 * you can use png_set_invalid(png_ptr, info_ptr, PNG_INFO_CHNK);
 */
PNG_EXPORT(177, void, png_set_invalid, (png_const_structrp png_ptr,
    png_inforp info_ptr, int mask));

#ifdef PNG_INFO_IMAGE_SUPPORTED
/* The "params" pointer is currently not used and is for future expansion. */
#ifdef PNG_SEQUENTIAL_READ_SUPPORTED
PNG_EXPORT(178, void, png_read_png, (png_structrp png_ptr, png_inforp info_ptr,
    int transforms, png_voidp params));
#endif
#ifdef PNG_WRITE_SUPPORTED
PNG_EXPORT(179, void, png_write_png, (png_structrp png_ptr, png_inforp info_ptr,
    int transforms, png_voidp params));
#endif
#endif

PNG_EXPORT(180, png_const_charp, png_get_copyright,
    (png_const_structrp png_ptr));
PNG_EXPORT(181, png_const_charp, png_get_header_ver,
    (png_const_structrp png_ptr));
PNG_EXPORT(182, png_const_charp, png_get_header_version,
    (png_const_structrp png_ptr));
PNG_EXPORT(183, png_const_charp, png_get_libpng_ver,
    (png_const_structrp png_ptr));

#ifdef PNG_MNG_FEATURES_SUPPORTED
PNG_EXPORT(184, png_uint_32, png_permit_mng_features, (png_structrp png_ptr,
```

```
   png_uint_32 mng_features_permitted));
#endif

/* For use in png_set_keep_unknown, added to version 1.2.6 */
#define PNG_HANDLE_CHUNK_AS_DEFAULT   0
#define PNG_HANDLE_CHUNK_NEVER        1
#define PNG_HANDLE_CHUNK_IF_SAFE      2
#define PNG_HANDLE_CHUNK_ALWAYS       3
#define PNG_HANDLE_CHUNK_LAST         4

/* Strip the prepended error numbers ("#nnn ") from error and warning
 * messages before passing them to the error or warning handler.
 */
#ifdef PNG_ERROR_NUMBERS_SUPPORTED
PNG_EXPORT(185, void, png_set_strip_error_numbers, (png_structrp png_ptr,
   png_uint_32 strip_mode));
#endif

/* Added in libpng-1.2.6 */
#ifdef PNG_SET_USER_LIMITS_SUPPORTED
PNG_EXPORT(186, void, png_set_user_limits, (png_structrp png_ptr,
   png_uint_32 user_width_max, png_uint_32 user_height_max));
PNG_EXPORT(187, png_uint_32, png_get_user_width_max,
   (png_const_structrp png_ptr));
PNG_EXPORT(188, png_uint_32, png_get_user_height_max,
   (png_const_structrp png_ptr));
/* Added in libpng-1.4.0 */
PNG_EXPORT(189, void, png_set_chunk_cache_max, (png_structrp png_ptr,
   png_uint_32 user_chunk_cache_max));
PNG_EXPORT(190, png_uint_32, png_get_chunk_cache_max,
   (png_const_structrp png_ptr));
/* Added in libpng-1.4.1 */
PNG_EXPORT(191, void, png_set_chunk_malloc_max, (png_structrp png_ptr,
   png_alloc_size_t user_chunk_cache_max));
PNG_EXPORT(192, png_alloc_size_t, png_get_chunk_malloc_max,
   (png_const_structrp png_ptr));
#endif

#if defined(PNG_INCH_CONVERSIONS_SUPPORTED)
PNG_EXPORT(193, png_uint_32, png_get_pixels_per_inch,
   (png_const_structrp png_ptr, png_const_inforp info_ptr));

PNG_EXPORT(194, png_uint_32, png_get_x_pixels_per_inch,
   (png_const_structrp png_ptr, png_const_inforp info_ptr));

PNG_EXPORT(195, png_uint_32, png_get_y_pixels_per_inch,
   (png_const_structrp png_ptr, png_const_inforp info_ptr));

PNG_FP_EXPORT(196, float, png_get_x_offset_inches,
   (png_const_structrp png_ptr, png_const_inforp info_ptr))
#ifdef PNG_FIXED_POINT_SUPPORTED /* otherwise not implemented. */
PNG_FIXED_EXPORT(211, png_fixed_point, png_get_x_offset_inches_fixed,
   (png_const_structrp png_ptr, png_const_inforp info_ptr))
#endif
```

```
PNG_FP_EXPORT(197, float, png_get_y_offset_inches, (png_const_structrp png_ptr,
   png_const_inforp info_ptr))
#ifdef PNG_FIXED_POINT_SUPPORTED /* otherwise not implemented. */
PNG_FIXED_EXPORT(212, png_fixed_point, png_get_y_offset_inches_fixed,
   (png_const_structrp png_ptr, png_const_inforp info_ptr))
#endif

#  ifdef PNG_pHYs_SUPPORTED
PNG_EXPORT(198, png_uint_32, png_get_pHYs_dpi, (png_const_structrp png_ptr,
   png_const_inforp info_ptr, png_uint_32 *res_x, png_uint_32 *res_y,
   int *unit_type));
#  endif /* PNG_pHYs_SUPPORTED */
#endif  /* PNG_INCH_CONVERSIONS_SUPPORTED */

/* Added in libpng-1.4.0 */
#ifdef PNG_IO_STATE_SUPPORTED
PNG_EXPORT(199, png_uint_32, png_get_io_state, (png_const_structrp png_ptr));

/* Removed from libpng 1.6; use png_get_io_chunk_type. */
PNG_REMOVED(200, png_const_bytep, png_get_io_chunk_name, (png_structrp png_ptr),
   PNG_DEPRECATED)

PNG_EXPORT(216, png_uint_32, png_get_io_chunk_type,
   (png_const_structrp png_ptr));

/* The flags returned by png_get_io_state() are the following: */
#  define PNG_IO_NONE       0x0000   /* no I/O at this moment */
#  define PNG_IO_READING    0x0001   /* currently reading */
#  define PNG_IO_WRITING    0x0002   /* currently writing */
#  define PNG_IO_SIGNATURE  0x0010   /* currently at the file signature */
#  define PNG_IO_CHUNK_HDR  0x0020   /* currently at the chunk header */
#  define PNG_IO_CHUNK_DATA 0x0040   /* currently at the chunk data */
#  define PNG_IO_CHUNK_CRC  0x0080   /* currently at the chunk crc */
#  define PNG_IO_MASK_OP    0x000f   /* current operation: reading/writing */
#  define PNG_IO_MASK_LOC   0x00f0   /* current location: sig/hdr/data/crc */
#endif /* ?PNG_IO_STATE_SUPPORTED */

/* Interlace support.  The following macros are always defined so that if
 * libpng interlace handling is turned off the macros may be used to handle
 * interlaced images within the application.
 */
#define PNG_INTERLACE_ADAM7_PASSES 7

/* Two macros to return the first row and first column of the original,
 * full, image which appears in a given pass.  'pass' is in the range 0
 * to 6 and the result is in the range 0 to 7.
 */
#define PNG_PASS_START_ROW(pass) (((1&~(pass))<<(3-((pass)>>1)))&7)
#define PNG_PASS_START_COL(pass) (((1& (pass))<<(3-(((pass)+1)>>1)))&7)

/* A macro to return the offset between pixels in the output row for a pair of
 * pixels in the input - effectively the inverse of the 'COL_SHIFT' macro that
 * follows.  Note that ROW_OFFSET is the offset from one row to the next whereas
```

```
 * COL_OFFSET is from one column to the next, within a row.
 */
#define PNG_PASS_ROW_OFFSET(pass) ((pass)>2?(8>>(((pass)-1)>>1)):8)
#define PNG_PASS_COL_OFFSET(pass) (1<<((7-(pass))>>1))

/* Two macros to help evaluate the number of rows or columns in each
 * pass.  This is expressed as a shift - effectively log2 of the number or
 * rows or columns in each 8x8 tile of the original image.
 */
#define PNG_PASS_ROW_SHIFT(pass) ((pass)>2?(8-(pass))>>1:3)
#define PNG_PASS_COL_SHIFT(pass) ((pass)>1?(7-(pass))>>1:3)

/* Hence two macros to determine the number of rows or columns in a given
 * pass of an image given its height or width.  In fact these macros may
 * return non-zero even though the sub-image is empty, because the other
 * dimension may be empty for a small image.
 */
#define PNG_PASS_ROWS(height, pass) (((height)+(((1<<PNG_PASS_ROW_SHIFT(pass))\
   -1)-PNG_PASS_START_ROW(pass)))>>PNG_PASS_ROW_SHIFT(pass))
#define PNG_PASS_COLS(width, pass) (((width)+(((1<<PNG_PASS_COL_SHIFT(pass))\
   -1)-PNG_PASS_START_COL(pass)))>>PNG_PASS_COL_SHIFT(pass))

/* For the reader row callbacks (both progressive and sequential) it is
 * necessary to find the row in the output image given a row in an interlaced
 * image, so two more macros:
 */
#define PNG_ROW_FROM_PASS_ROW(y_in, pass) \
   (((y_in)<<PNG_PASS_ROW_SHIFT(pass))+PNG_PASS_START_ROW(pass))
#define PNG_COL_FROM_PASS_COL(x_in, pass) \
   (((x_in)<<PNG_PASS_COL_SHIFT(pass))+PNG_PASS_START_COL(pass))

/* Two macros which return a boolean (0 or 1) saying whether the given row
 * or column is in a particular pass.  These use a common utility macro that
 * returns a mask for a given pass - the offset 'off' selects the row or
 * column version.  The mask has the appropriate bit set for each column in
 * the tile.
 */
#define PNG_PASS_MASK(pass,off) ( \
   ((0x110145AF>>(((7-(off))-(pass))<<2)) & 0xF) | \
   ((0x01145AF0>>(((7-(off))-(pass))<<2)) & 0xF0))

#define PNG_ROW_IN_INTERLACE_PASS(y, pass) \
   ((PNG_PASS_MASK(pass,0) >> ((y)&7)) & 1)
#define PNG_COL_IN_INTERLACE_PASS(x, pass) \
   ((PNG_PASS_MASK(pass,1) >> ((x)&7)) & 1)

#ifdef PNG_READ_COMPOSITE_NODIV_SUPPORTED
/* With these routines we avoid an integer divide, which will be slower on
 * most machines.  However, it does take more operations than the corresponding
 * divide method, so it may be slower on a few RISC systems.  There are two
 * shifts (by 8 or 16 bits) and an addition, versus a single integer divide.
 *
 * Note that the rounding factors are NOT supposed to be the same!  128 and
 * 32768 are correct for the NODIV code; 127 and 32767 are correct for the
```

```
 * standard method.
 *
 * [Optimized code by Greg Roelofs and Mark Adler...blame us for bugs. :-) ]
 */

 /* fg and bg should be in `gamma 1.0' space; alpha is the opacity */

#  define png_composite(composite, fg, alpha, bg)         \
    { png_uint_16 temp = (png_uint_16)((png_uint_16)(fg) \
        * (png_uint_16)(alpha)                           \
        + (png_uint_16)(bg)*(png_uint_16)(255            \
        - (png_uint_16)(alpha)) + 128);                  \
      (composite) = (png_byte)((temp + (temp >> 8)) >> 8); }

#  define png_composite_16(composite, fg, alpha, bg)       \
    { png_uint_32 temp = (png_uint_32)((png_uint_32)(fg)  \
        * (png_uint_32)(alpha)                           \
        + (png_uint_32)(bg)*(65535                       \
        - (png_uint_32)(alpha)) + 32768);                \
      (composite) = (png_uint_16)((temp + (temp >> 16)) >> 16); }

#else  /* Standard method using integer division */

#  define png_composite(composite, fg, alpha, bg)                          \
    (composite) = (png_byte)(((png_uint_16)(fg) * (png_uint_16)(alpha) +  \
    (png_uint_16)(bg) * (png_uint_16)(255 - (png_uint_16)(alpha)) +       \
    127) / 255)

#  define png_composite_16(composite, fg, alpha, bg)                       \
    (composite) = (png_uint_16)(((png_uint_32)(fg) * (png_uint_32)(alpha) + \
    (png_uint_32)(bg)*(png_uint_32)(65535 - (png_uint_32)(alpha)) +         \
    32767) / 65535)
#endif /* PNG_READ_COMPOSITE_NODIV_SUPPORTED */

#ifdef PNG_READ_INT_FUNCTIONS_SUPPORTED
PNG_EXPORT(201, png_uint_32, png_get_uint_32, (png_const_bytep buf));
PNG_EXPORT(202, png_uint_16, png_get_uint_16, (png_const_bytep buf));
PNG_EXPORT(203, png_int_32, png_get_int_32, (png_const_bytep buf));
#endif

PNG_EXPORT(204, png_uint_32, png_get_uint_31, (png_const_structrp png_ptr,
   png_const_bytep buf));
/* No png_get_int_16 -- may be added if there's a real need for it. */

/* Place a 32-bit number into a buffer in PNG byte order (big-endian). */
#ifdef PNG_WRITE_INT_FUNCTIONS_SUPPORTED
PNG_EXPORT(205, void, png_save_uint_32, (png_bytep buf, png_uint_32 i));
#endif
#ifdef PNG_SAVE_INT_32_SUPPORTED
PNG_EXPORT(206, void, png_save_int_32, (png_bytep buf, png_int_32 i));
#endif

/* Place a 16-bit number into a buffer in PNG byte order.
 * The parameter is declared unsigned int, not png_uint_16,
```

```
 * just to avoid potential problems on pre-ANSI C compilers.
 */
#ifdef PNG_WRITE_INT_FUNCTIONS_SUPPORTED
PNG_EXPORT(207, void, png_save_uint_16, (png_bytep buf, unsigned int i));
/* No png_save_int_16 -- may be added if there's a real need for it. */
#endif

#ifdef PNG_USE_READ_MACROS
/* Inline macros to do direct reads of bytes from the input buffer.
 * The png_get_int_32() routine assumes we are using two's complement
 * format for negative values, which is almost certainly true.
 */
#  define PNG_get_uint_32(buf) \
    (((png_uint_32)(*(buf)) << 24) + \
     ((png_uint_32)(*((buf) + 1)) << 16) + \
     ((png_uint_32)(*((buf) + 2)) << 8) + \
     ((png_uint_32)(*((buf) + 3))))

   /* From libpng-1.4.0 until 1.4.4, the png_get_uint_16 macro (but not the
    * function) incorrectly returned a value of type png_uint_32.
    */
#  define PNG_get_uint_16(buf) \
    ((png_uint_16) \
     (((unsigned int)(*(buf)) << 8) + \
      ((unsigned int)(*((buf) + 1)))))

#  define PNG_get_int_32(buf) \
    ((png_int_32)((*(buf) & 0x80) \
     ? -((png_int_32)((png_get_uint_32(buf) ^ 0xffffffffL) + 1)) \
     : (png_int_32)png_get_uint_32(buf)))

   /* If PNG_PREFIX is defined the same thing as below happens in pnglibconf.h,
    * but defining a macro name prefixed with PNG_PREFIX.
    */
#  ifndef PNG_PREFIX
#    define png_get_uint_32(buf) PNG_get_uint_32(buf)
#    define png_get_uint_16(buf) PNG_get_uint_16(buf)
#    define png_get_int_32(buf)  PNG_get_int_32(buf)
#  endif
#else
#  ifdef PNG_PREFIX
     /* No macros; revert to the (redefined) function */
#    define PNG_get_uint_32 (png_get_uint_32)
#    define PNG_get_uint_16 (png_get_uint_16)
#    define PNG_get_int_32  (png_get_int_32)
#  endif
#endif

/*******************************************************************************
 *  SIMPLIFIED API
 *******************************************************************************
 *
 * Please read the documentation in libpng-manual.txt (TODO: write said
 * documentation) if you don't understand what follows.
```

```
 *
 * The simplified API hides the details of both libpng and the PNG file format
 * itself.  It allows PNG files to be read into a very limited number of
 * in-memory bitmap formats or to be written from the same formats.  If these
 * formats do not accomodate your needs then you can, and should, use the more
 * sophisticated APIs above - these support a wide variety of in-memory formats
 * and a wide variety of sophisticated transformations to those formats as well
 * as a wide variety of APIs to manipulate ancillary information.
 *
 * To read a PNG file using the simplified API:
 *
 * 1) Declare a 'png_image' structure (see below) on the stack and set the
 *    version field to PNG_IMAGE_VERSION.
 * 2) Call the appropriate png_image_begin_read... function.
 * 3) Set the png_image 'format' member to the required sample format.
 * 4) Allocate a buffer for the image and, if required, the color-map.
 * 5) Call png_image_finish_read to read the image and, if required, the
 *    color-map into your buffers.
 *
 * There are no restrictions on the format of the PNG input itself; all valid
 * color types, bit depths, and interlace methods are acceptable, and the
 * input image is transformed as necessary to the requested in-memory format
 * during the png_image_finish_read() step.  The only caveat is that if you
 * request a color-mapped image from a PNG that is full-color or makes
 * complex use of an alpha channel the transformation is extremely lossy and the
 * result may look terrible.
 *
 * To write a PNG file using the simplified API:
 *
 * 1) Declare a 'png_image' structure on the stack and memset() it to all zero.
 * 2) Initialize the members of the structure that describe the image, setting
 *    the 'format' member to the format of the image samples.
 * 3) Call the appropriate png_image_write... function with a pointer to the
 *    image and, if necessary, the color-map to write the PNG data.
 *
 * png_image is a structure that describes the in-memory format of an image
 * when it is being read or defines the in-memory format of an image that you
 * need to write:
 */
#define PNG_IMAGE_VERSION 1

typedef struct png_control *png_controlp;
typedef struct
{
   png_controlp opaque;    /* Initialize to NULL, free with png_image_free */
   png_uint_32  version;   /* Set to PNG_IMAGE_VERSION */
   png_uint_32  width;     /* Image width in pixels (columns) */
   png_uint_32  height;    /* Image height in pixels (rows) */
   png_uint_32  format;    /* Image format as defined below */
   png_uint_32  flags;     /* A bit mask containing informational flags */
   png_uint_32  colormap_entries;
                       /* Number of entries in the color-map */

   /* In the event of an error or warning the following field will be set to a
```

```
    * non-zero value and the 'message' field will contain a '\0' terminated
    * string with the libpng error or warning message.  If both warnings and
    * an error were encountered, only the error is recorded.  If there
    * are multiple warnings, only the first one is recorded.
    *
    * The upper 30 bits of this value are reserved, the low two bits contain
    * a value as follows:
    */
#  define PNG_IMAGE_WARNING 1
#  define PNG_IMAGE_ERROR 2
   /*
    * The result is a two bit code such that a value more than 1 indicates
    * a failure in the API just called:
    *
    *    0 - no warning or error
    *    1 - warning
    *    2 - error
    *    3 - error preceded by warning
    */
#  define PNG_IMAGE_FAILED(png_cntrl) ((((png_cntrl).warning_or_error)&0x03)>1)

   png_uint_32  warning_or_error;

   char        message[64];
} png_image, *png_imagep;

/* The samples of the image have one to four channels whose components have
 * original values in the range 0 to 1.0:
 *
 * 1: A single gray or luminance channel (G).
 * 2: A gray/luminance channel and an alpha channel (GA).
 * 3: Three red, green, blue color channels (RGB).
 * 4: Three color channels and an alpha channel (RGBA).
 *
 * The components are encoded in one of two ways:
 *
 * a) As a small integer, value 0..255, contained in a single byte.  For the
 * alpha channel the original value is simply value/255.  For the color or
 * luminance channels the value is encoded according to the sRGB specification
 * and matches the 8-bit format expected by typical display devices.
 *
 * The color/gray channels are not scaled (pre-multiplied) by the alpha
 * channel and are suitable for passing to color management software.
 *
 * b) As a value in the range 0..65535, contained in a 2-byte integer.  All
 * channels can be converted to the original value by dividing by 65535; all
 * channels are linear.  Color channels use the RGB encoding (RGB end-points) of
 * the sRGB specification.  This encoding is identified by the
 * PNG_FORMAT_FLAG_LINEAR flag below.
 *
 * When the simplified API needs to convert between sRGB and linear colorspaces,
 * the actual sRGB transfer curve defined in the sRGB specification (see the
 * article at http://en.wikipedia.org/wiki/SRGB) is used, not the gamma=1/2.2
 * approximation used elsewhere in libpng.
```

```
 *
 * When an alpha channel is present it is expected to denote pixel coverage
 * of the color or luminance channels and is returned as an associated alpha
 * channel: the color/gray channels are scaled (pre-multiplied) by the alpha
 * value.
 *
 * The samples are either contained directly in the image data, between 1 and 8
 * bytes per pixel according to the encoding, or are held in a color-map indexed
 * by bytes in the image data.  In the case of a color-map the color-map entries
 * are individual samples, encoded as above, and the image data has one byte per
 * pixel to select the relevant sample from the color-map.
 */

/* PNG_FORMAT_*
 *
 * #defines to be used in png_image::format.  Each #define identifies a
 * particular layout of sample data and, if present, alpha values.  There are
 * separate defines for each of the two component encodings.
 *
 * A format is built up using single bit flag values.  All combinations are
 * valid.  Formats can be built up from the flag values or you can use one of
 * the predefined values below.  When testing formats always use the FORMAT_FLAG
 * macros to test for individual features - future versions of the library may
 * add new flags.
 *
 * When reading or writing color-mapped images the format should be set to the
 * format of the entries in the color-map then png_image_{read,write}_colormap
 * called to read or write the color-map and set the format correctly for the
 * image data.  Do not set the PNG_FORMAT_FLAG_COLORMAP bit directly!
 *
 * NOTE: libpng can be built with particular features disabled, if you see
 * compiler errors because the definition of one of the following flags has been
 * compiled out it is because libpng does not have the required support.  It is
 * possible, however, for the libpng configuration to enable the format on just
 * read or just write; in that case you may see an error at run time.  You can
 * guard against this by checking for the definition of the appropriate
 * "_SUPPORTED" macro, one of:
 *
 *    PNG_SIMPLIFIED_{READ,WRITE}_{BGR,AFIRST}_SUPPORTED
 */
#define PNG_FORMAT_FLAG_ALPHA    0x01U /* format with an alpha channel */
#define PNG_FORMAT_FLAG_COLOR    0x02U /* color format: otherwise grayscale */
#define PNG_FORMAT_FLAG_LINEAR   0x04U /* 2 byte channels else 1 byte */
#define PNG_FORMAT_FLAG_COLORMAP 0x08U /* image data is color-mapped */

#ifdef PNG_FORMAT_BGR_SUPPORTED
#  define PNG_FORMAT_FLAG_BGR    0x10U /* BGR colors, else order is RGB */
#endif

#ifdef PNG_FORMAT_AFIRST_SUPPORTED
#  define PNG_FORMAT_FLAG_AFIRST 0x20U /* alpha channel comes first */
#endif

/* Commonly used formats have predefined macros.
```

```
 *
 * First the single byte (sRGB) formats:
 */
#define PNG_FORMAT_GRAY 0
#define PNG_FORMAT_GA   PNG_FORMAT_FLAG_ALPHA
#define PNG_FORMAT_AG   (PNG_FORMAT_GA|PNG_FORMAT_FLAG_AFIRST)
#define PNG_FORMAT_RGB  PNG_FORMAT_FLAG_COLOR
#define PNG_FORMAT_BGR  (PNG_FORMAT_FLAG_COLOR|PNG_FORMAT_FLAG_BGR)
#define PNG_FORMAT_RGBA (PNG_FORMAT_RGB|PNG_FORMAT_FLAG_ALPHA)
#define PNG_FORMAT_ARGB (PNG_FORMAT_RGBA|PNG_FORMAT_FLAG_AFIRST)
#define PNG_FORMAT_BGRA (PNG_FORMAT_BGR|PNG_FORMAT_FLAG_ALPHA)
#define PNG_FORMAT_ABGR (PNG_FORMAT_BGRA|PNG_FORMAT_FLAG_AFIRST)

/* Then the linear 2-byte formats.  When naming these "Y" is used to
 * indicate a luminance (gray) channel.
 */
#define PNG_FORMAT_LINEAR_Y PNG_FORMAT_FLAG_LINEAR
#define PNG_FORMAT_LINEAR_Y_ALPHA (PNG_FORMAT_FLAG_LINEAR|PNG_FORMAT_FLAG_ALPHA)
#define PNG_FORMAT_LINEAR_RGB (PNG_FORMAT_FLAG_LINEAR|PNG_FORMAT_FLAG_COLOR)
#define PNG_FORMAT_LINEAR_RGB_ALPHA \
   (PNG_FORMAT_FLAG_LINEAR|PNG_FORMAT_FLAG_COLOR|PNG_FORMAT_FLAG_ALPHA)

/* With color-mapped formats the image data is one byte for each pixel, the byte
 * is an index into the color-map which is formatted as above.  To obtain a
 * color-mapped format it is sufficient just to add the PNG_FOMAT_FLAG_COLORMAP
 * to one of the above definitions, or you can use one of the definitions below.
 */
#define PNG_FORMAT_RGB_COLORMAP  (PNG_FORMAT_RGB|PNG_FORMAT_FLAG_COLORMAP)
#define PNG_FORMAT_BGR_COLORMAP  (PNG_FORMAT_BGR|PNG_FORMAT_FLAG_COLORMAP)
#define PNG_FORMAT_RGBA_COLORMAP (PNG_FORMAT_RGBA|PNG_FORMAT_FLAG_COLORMAP)
#define PNG_FORMAT_ARGB_COLORMAP (PNG_FORMAT_ARGB|PNG_FORMAT_FLAG_COLORMAP)
#define PNG_FORMAT_BGRA_COLORMAP (PNG_FORMAT_BGRA|PNG_FORMAT_FLAG_COLORMAP)
#define PNG_FORMAT_ABGR_COLORMAP (PNG_FORMAT_ABGR|PNG_FORMAT_FLAG_COLORMAP)

/* PNG_IMAGE macros
 *
 * These are convenience macros to derive information from a png_image
 * structure.  The PNG_IMAGE_SAMPLE_ macros return values appropriate to the
 * actual image sample values - either the entries in the color-map or the
 * pixels in the image.  The PNG_IMAGE_PIXEL_ macros return corresponding values
 * for the pixels and will always return 1 for color-mapped formats.  The
 * remaining macros return information about the rows in the image and the
 * complete image.
 *
 * NOTE: All the macros that take a png_image::format parameter are compile time
 * constants if the format parameter is, itself, a constant.  Therefore these
 * macros can be used in array declarations and case labels where required.
 * Similarly the macros are also pre-processor constants (sizeof is not used) so
 * they can be used in #if tests.
 *
 * First the information about the samples.
 */
#define PNG_IMAGE_SAMPLE_CHANNELS(fmt)\
   (((fmt)&(PNG_FORMAT_FLAG_COLOR|PNG_FORMAT_FLAG_ALPHA))+1)
```

```
   /* Return the total number of channels in a given format: 1..4 */

#define PNG_IMAGE_SAMPLE_COMPONENT_SIZE(fmt)\
   ((((fmt) & PNG_FORMAT_FLAG_LINEAR) >> 2)+1)
   /* Return the size in bytes of a single component of a pixel or color-map
    * entry (as appropriate) in the image: 1 or 2.
    */

#define PNG_IMAGE_SAMPLE_SIZE(fmt)\
   (PNG_IMAGE_SAMPLE_CHANNELS(fmt) * PNG_IMAGE_SAMPLE_COMPONENT_SIZE(fmt))
   /* This is the size of the sample data for one sample.  If the image is
    * color-mapped it is the size of one color-map entry (and image pixels are
    * one byte in size), otherwise it is the size of one image pixel.
    */

#define PNG_IMAGE_MAXIMUM_COLORMAP_COMPONENTS(fmt)\
   (PNG_IMAGE_SAMPLE_CHANNELS(fmt) * 256)
   /* The maximum size of the color-map required by the format expressed in a
    * count of components.  This can be used to compile-time allocate a
    * color-map:
    *
    * png_uint_16 colormap[PNG_IMAGE_MAXIMUM_COLORMAP_COMPONENTS(linear_fmt)];
    *
    * png_byte colormap[PNG_IMAGE_MAXIMUM_COLORMAP_COMPONENTS(sRGB_fmt)];
    *
    * Alternatively use the PNG_IMAGE_COLORMAP_SIZE macro below to use the
    * information from one of the png_image_begin_read_ APIs and dynamically
    * allocate the required memory.
    */

/* Corresponding information about the pixels */
#define PNG_IMAGE_PIXEL_(test,fmt)\
   (((fmt)&PNG_FORMAT_FLAG_COLORMAP)?1:test(fmt))

#define PNG_IMAGE_PIXEL_CHANNELS(fmt)\
   PNG_IMAGE_PIXEL_(PNG_IMAGE_SAMPLE_CHANNELS,fmt)
   /* The number of separate channels (components) in a pixel; 1 for a
    * color-mapped image.
    */

#define PNG_IMAGE_PIXEL_COMPONENT_SIZE(fmt)\
   PNG_IMAGE_PIXEL_(PNG_IMAGE_SAMPLE_COMPONENT_SIZE,fmt)
   /* The size, in bytes, of each component in a pixel; 1 for a color-mapped
    * image.
    */

#define PNG_IMAGE_PIXEL_SIZE(fmt) PNG_IMAGE_PIXEL_(PNG_IMAGE_SAMPLE_SIZE,fmt)
   /* The size, in bytes, of a complete pixel; 1 for a color-mapped image. */

/* Information about the whole row, or whole image */
#define PNG_IMAGE_ROW_STRIDE(image)\
   (PNG_IMAGE_PIXEL_CHANNELS((image).format) * (image).width)
   /* Return the total number of components in a single row of the image; this
    * is the minimum 'row stride', the minimum count of components between each
```

```
    * row.  For a color-mapped image this is the minimum number of bytes in a
    * row.
    */

#define PNG_IMAGE_BUFFER_SIZE(image, row_stride)\
   (PNG_IMAGE_PIXEL_COMPONENT_SIZE((image).format)*(image).height*(row_stride))
   /* Return the size, in bytes, of an image buffer given a png_image and a row
    * stride - the number of components to leave space for in each row.
    */

#define PNG_IMAGE_SIZE(image)\
   PNG_IMAGE_BUFFER_SIZE(image, PNG_IMAGE_ROW_STRIDE(image))
   /* Return the size, in bytes, of the image in memory given just a png_image;
    * the row stride is the minimum stride required for the image.
    */

#define PNG_IMAGE_COLORMAP_SIZE(image)\
   (PNG_IMAGE_SAMPLE_SIZE((image).format) * (image).colormap_entries)
   /* Return the size, in bytes, of the color-map of this image.  If the image
    * format is not a color-map format this will return a size sufficient for
    * 256 entries in the given format; check PNG_FORMAT_FLAG_COLORMAP if
    * you don't want to allocate a color-map in this case.
    */

/* PNG_IMAGE_FLAG_*
 *
 * Flags containing additional information about the image are held in the
 * 'flags' field of png_image.
 */
#define PNG_IMAGE_FLAG_COLORSPACE_NOT_sRGB 0x01
   /* This indicates the the RGB values of the in-memory bitmap do not
    * correspond to the red, green and blue end-points defined by sRGB.
    */

#define PNG_IMAGE_FLAG_FAST 0x02
   /* On write emphasise speed over compression; the resultant PNG file will be
    * larger but will be produced significantly faster, particular for large
    * images.  Do not use this option for images which will be distributed, only
    * used it when producing intermediate files that will be read back in
    * repeatedly.  For a typical 24-bit image the option will double the read
    * speed at the cost of increasing the image size by 25%, however for many
    * more compressible images the PNG file can be 10 times larger with only a
    * slight speed gain.
    */

#define PNG_IMAGE_FLAG_16BIT_sRGB 0x04
   /* On read if the image is a 16-bit per component image and there is no gAMA
    * or sRGB chunk assume that the components are sRGB encoded.  Notice that
    * images output by the simplified API always have gamma information; setting
    * this flag only affects the interpretation of 16-bit images from an
    * external source.  It is recommended that the application expose this flag
    * to the user; the user can normally easily recognize the difference between
    * linear and sRGB encoding.  This flag has no effect on write - the data
    * passed to the write APIs must have the correct encoding (as defined
```

```
    * above.)
    *
    * If the flag is not set (the default) input 16-bit per component data is
    * assumed to be linear.
    *
    * NOTE: the flag can only be set after the png_image_begin_read_ call,
    * because that call initializes the 'flags' field.
    */

#ifdef PNG_SIMPLIFIED_READ_SUPPORTED
/* READ APIs
 * ---------
 *
 * The png_image passed to the read APIs must have been initialized by setting
 * the png_controlp field 'opaque' to NULL (or, safer, memset the whole thing.)
 */
#ifdef PNG_STDIO_SUPPORTED
PNG_EXPORT(234, int, png_image_begin_read_from_file, (png_imagep image,
    const char *file_name));
    /* The named file is opened for read and the image header is filled in
     * from the PNG header in the file.
     */

PNG_EXPORT(235, int, png_image_begin_read_from_stdio, (png_imagep image,
    FILE* file));
    /* The PNG header is read from the stdio FILE object. */
#endif /* PNG_STDIO_SUPPORTED */

PNG_EXPORT(236, int, png_image_begin_read_from_memory, (png_imagep image,
    png_const_voidp memory, png_size_t size));
    /* The PNG header is read from the given memory buffer. */

PNG_EXPORT(237, int, png_image_finish_read, (png_imagep image,
    png_const_colorp background, void *buffer, png_int_32 row_stride,
    void *colormap));
    /* Finish reading the image into the supplied buffer and clean up the
     * png_image structure.
     *
     * row_stride is the step, in byte or 2-byte units as appropriate,
     * between adjacent rows.  A positive stride indicates that the top-most row
     * is first in the buffer - the normal top-down arrangement.  A negative
     * stride indicates that the bottom-most row is first in the buffer.
     *
     * background need only be supplied if an alpha channel must be removed from
     * a png_byte format and the removal is to be done by compositing on a solid
     * color; otherwise it may be NULL and any composition will be done directly
     * onto the buffer.  The value is an sRGB color to use for the background,
     * for grayscale output the green channel is used.
     *
     * background must be supplied when an alpha channel must be removed from a
     * single byte color-mapped output format, in other words if:
     *
     * 1) The original format from png_image_begin_read_from_* had
     *    PNG_FORMAT_FLAG_ALPHA set.
```

```
    * 2) The format set by the application does not.
    * 3) The format set by the application has PNG_FORMAT_FLAG_COLORMAP set and
    *    PNG_FORMAT_FLAG_LINEAR *not* set.
    *
    * For linear output removing the alpha channel is always done by compositing
    * on black and background is ignored.
    *
    * colormap must be supplied when PNG_FORMAT_FLAG_COLORMAP is set.  It must
    * be at least the size (in bytes) returned by PNG_IMAGE_COLORMAP_SIZE.
    * image->colormap_entries will be updated to the actual number of entries
    * written to the colormap; this may be less than the original value.
    */

  PNG_EXPORT(238, void, png_image_free, (png_imagep image));
    /* Free any data allocated by libpng in image->opaque, setting the pointer to
    * NULL.  May be called at any time after the structure is initialized.
    */
  #endif /* PNG_SIMPLIFIED_READ_SUPPORTED */

  #ifdef PNG_SIMPLIFIED_WRITE_SUPPORTED
  #ifdef PNG_STDIO_SUPPORTED
  /* WRITE APIS
   * ----------
   * For write you must initialize a png_image structure to describe the image to
   * be written.  To do this use memset to set the whole structure to 0 then
   * initialize fields describing your image.
   *
   * version: must be set to PNG_IMAGE_VERSION
   * opaque: must be initialized to NULL
   * width: image width in pixels
   * height: image height in rows
   * format: the format of the data (image and color-map) you wish to write
   * flags: set to 0 unless one of the defined flags applies; set
   *    PNG_IMAGE_FLAG_COLORSPACE_NOT_sRGB for color format images where the RGB
   *    values do not correspond to the colors in sRGB.
   * colormap_entries: set to the number of entries in the color-map (0 to 256)
   */
  PNG_EXPORT(239, int, png_image_write_to_file, (png_imagep image,
    const char *file, int convert_to_8bit, const void *buffer,
    png_int_32 row_stride, const void *colormap));
    /* Write the image to the named file. */

  PNG_EXPORT(240, int, png_image_write_to_stdio, (png_imagep image, FILE *file,
    int convert_to_8_bit, const void *buffer, png_int_32 row_stride,
    const void *colormap));
    /* Write the image to the given (FILE*). */

  /* With both write APIs if image is in one of the linear formats with 16-bit
   * data then setting convert_to_8_bit will cause the output to be an 8-bit PNG
   * gamma encoded according to the sRGB specification, otherwise a 16-bit linear
   * encoded PNG file is written.
   *
   * With color-mapped data formats the colormap parameter point to a color-map
   * with at least image->colormap_entries encoded in the specified format.  If
```

```
 * the format is linear the written PNG color-map will be converted to sRGB
 * regardless of the convert_to_8_bit flag.
 *
 * With all APIs row_stride is handled as in the read APIs - it is the spacing
 * from one row to the next in component sized units (1 or 2 bytes) and if
 * negative indicates a bottom-up row layout in the buffer.
 *
 * Note that the write API does not support interlacing or sub-8-bit pixels.
 */
#endif /* PNG_STDIO_SUPPORTED */
#endif /* PNG_SIMPLIFIED_WRITE_SUPPORTED */
/*******************************************************************************
 *  END OF SIMPLIFIED API
 ******************************************************************************/

#ifdef PNG_CHECK_FOR_INVALID_INDEX_SUPPORTED
PNG_EXPORT(242, void, png_set_check_for_invalid_index,
   (png_structrp png_ptr, int allowed));
#  ifdef PNG_GET_PALETTE_MAX_SUPPORTED
PNG_EXPORT(243, int, png_get_palette_max, (png_const_structp png_ptr,
   png_const_infop info_ptr));
#  endif
#endif /* CHECK_FOR_INVALID_INDEX */

/*******************************************************************************
 *  IMPLEMENTATION OPTIONS
 *******************************************************************************
 *
 * Support for arbitrary implementation-specific optimizations.  The API allows
 * particular options to be turned on or off.  'Option' is the number of the
 * option and 'onoff' is 0 (off) or non-0 (on).  The value returned is given
 * by the PNG_OPTION_ defines below.
 *
 * HARDWARE: normally hardware capabilites, such as the Intel SSE instructions,
 *      are detected at run time, however sometimes it may be impossible
 *      to do this in user mode, in which case it is necessary to discover
 *      the capabilities in an OS specific way.  Such capabilities are
 *      listed here when libpng has support for them and must be turned
 *      ON by the application if present.
 *
 * SOFTWARE: sometimes software optimizations actually result in performance
 *      decrease on some architectures or systems, or with some sets of
 *      PNG images.  'Software' options allow such optimizations to be
 *      selected at run time.
 */
#ifdef PNG_SET_OPTION_SUPPORTED
#ifdef PNG_ARM_NEON_API_SUPPORTED
#  define PNG_ARM_NEON   0 /* HARDWARE: ARM Neon SIMD instructions supported */
#endif
#define PNG_MAXIMUM_INFLATE_WINDOW 2 /* SOFTWARE: force maximum window */
#define PNG_SKIP_sRGB_CHECK_PROFILE 4 /* SOFTWARE: Check ICC profile for sRGB */
#define PNG_OPTION_NEXT  6 /* Next option - numbers must be even */

/* Return values: NOTE: there are four values and 'off' is *not* zero */
```

```
#define PNG_OPTION_UNSET   0 /* Unset - defaults to off */
#define PNG_OPTION_INVALID 1 /* Option number out of range */
#define PNG_OPTION_OFF     2
#define PNG_OPTION_ON      3

PNG_EXPORT(244, int, png_set_option, (png_structrp png_ptr, int option,
   int onoff));
#endif

/*************************************************************************
 *  END OF HARDWARE AND SOFTWARE OPTIONS
 *************************************************************************/

/* Maintainer: Put new public prototypes here ^, in libpng.3, and project
 * defs, scripts/pnglibconf.h, and scripts/pnglibconf.h.prebuilt
 */

/* The last ordinal number (this is the *last* one already used; the next
 * one to use is one more than this.)  Maintainer, remember to add an entry to
 * scripts/symbols.def as well.
 */
#ifdef PNG_EXPORT_LAST_ORDINAL
  PNG_EXPORT_LAST_ORDINAL(244);
#endif

#ifdef __cplusplus
}
#endif

#endif /* PNG_VERSION_INFO_ONLY */
/* Do not put anything past this line */
#endif /* PNG_H */
```

The Independent JPEG Group's JPEG software
==========

README for release 8d of 15-Jan-2012
==========

This distribution contains the eighth public release of the Independent JPEG Group's free JPEG software. You are welcome to redistribute this software and to use it for any purpose, subject to the conditions under LEGAL ISSUES, below.

This software is the work of Tom Lane, Guido Vollbeding, Philip Gladstone, Bill Allombert, Jim Boucher, Lee Crocker, Bob Friesenhahn, Ben Jackson, Julian Minguillon, Luis Ortiz, George Phillips, Davide Rossi, Ge' Weijers, and other members of the Independent JPEG Group.

IJG is not affiliated with the ISO/IEC JTC1/SC29/WG1 standards committee (also known as JPEG, together with ITU-T SG16).

DOCUMENTATION ROADMAP
==========

This file contains the following sections:



Other documentation files in the distribution are:

User documentation:
- install.txt How to configure and install the IJG software.
- usage.txt Usage instructions for cjpeg, djpeg, jpegtran, rdjpgcom, and wrjpgcom.
- *.1 Unix-style man pages for programs (same info as usage.txt).
- wizard.txt Advanced usage instructions for JPEG wizards only.
- change.log Version-to-version change highlights.

Programmer and internal documentation:
- libjpeg.txt How to use the JPEG library in your own programs.
- example.c Sample code for calling the JPEG library.
- structure.txt Overview of the JPEG library's internal structure.
- filelist.txt Road map of IJG files.
- coderules.txt Coding style rules --- please read if you contribute code.

Please read at least the files install.txt and usage.txt. Some information can also be found in the JPEG FAQ (Frequently Asked Questions) article. See ARCHIVE LOCATIONS below to find out where to obtain the FAQ article.

If you want to understand how the JPEG code works, we suggest reading one or

more of the REFERENCES, then looking at the documentation files (in roughly the order listed) before diving into the code.

## OVERVIEW

This package contains C software to implement JPEG image encoding, decoding, and transcoding. JPEG (pronounced "jay-peg") is a standardized compression method for full-color and gray-scale images.

This software implements JPEG baseline, extended-sequential, and progressive compression processes. Provision is made for supporting all variants of these processes, although some uncommon parameter settings aren't implemented yet. We have made no provision for supporting the hierarchical or lossless processes defined in the standard.

We provide a set of library routines for reading and writing JPEG image files, plus two sample applications "cjpeg" and "djpeg", which use the library to perform conversion between JPEG and some other popular image file formats. The library is intended to be reused in other applications.

In order to support file conversion and viewing software, we have included considerable functionality beyond the bare JPEG coding/decoding capability; for example, the color quantization modules are not strictly part of JPEG decoding, but they are essential for output to colormapped file formats or colormapped displays. These extra functions can be compiled out of the library if not required for a particular application.

We have also included "jpegtran", a utility for lossless transcoding between different JPEG processes, and "rdjpgcom" and "wrjpgcom", two simple applications for inserting and extracting textual comments in JFIF files.

The emphasis in designing this software has been on achieving portability and flexibility, while also making it fast enough to be useful. In particular, the software is not intended to be read as a tutorial on JPEG. (See the REFERENCES section for introductory material.) Rather, it is intended to be reliable, portable, industrial-strength code. We do not claim to have achieved that goal in every aspect of the software, but we strive for it.

We welcome the use of this software as a component of commercial products. No royalty is required, but we do ask for an acknowledgement in product documentation, as described under LEGAL ISSUES.

## LEGAL ISSUES

In plain English:

1. We don't promise that this software works. (But if you find any bugs, please let us know!)
2. You can use this software for whatever you want. You don't have to pay us.
3. You may not pretend that you wrote this software. If you use it in a

program, you must acknowledge somewhere in your documentation that
you've used the IJG code.

In legalese:

The authors make NO WARRANTY or representation, either express or implied,
with respect to this software, its quality, accuracy, merchantability, or
fitness for a particular purpose.  This software is provided "AS IS", and you,
its user, assume the entire risk as to its quality and accuracy.



Permission is hereby granted to use, copy, modify, and distribute this
software (or portions thereof) for any purpose, without fee, subject to these
conditions:
(1) If any part of the source code for this software is distributed, then this
README file must be included, with this copyright and no-warranty notice
unaltered; and any additions, deletions, or changes to the original files
must be clearly indicated in accompanying documentation.
(2) If only executable code is distributed, then the accompanying
documentation must state that "this software is based in part on the work of
the Independent JPEG Group".
(3) Permission for use of this software is granted only if the user accepts
full responsibility for any undesirable consequences; the authors accept
NO LIABILITY for damages of any kind.

These conditions apply to any software derived from or based on the IJG code,
not just to the unmodified library.  If you use our work, you ought to
acknowledge us.

Permission is NOT granted for the use of any IJG author's name or company name
in advertising or publicity relating to this software or products derived from
it.  This software may be referred to only as "the Independent JPEG Group's
software".

We specifically permit and encourage the use of this software as the basis of
commercial products, provided that all warranty or liability claims are
assumed by the product vendor.


ansi2knr.c is included in this distribution by permission of L. Peter Deutsch,
sole proprietor of its copyright holder, Aladdin Enterprises of Menlo Park, CA.
ansi2knr.c is NOT covered by the above copyright and conditions, but instead
by the usual distribution terms of the Free Software Foundation; principally,
that you must include source code if you redistribute it.  (See the file
ansi2knr.c for full details.)  However, since ansi2knr.c is not needed as part
of any program generated from the IJG code, this does not limit you more than
the foregoing paragraphs do.

The Unix configuration script "configure" was produced with GNU Autoconf.
It is copyright by the Free Software Foundation but is freely distributable.
The same holds for its supporting scripts (config.guess, config.sub,
ltmain.sh).  Another support script, install-sh, is copyright by X Consortium

but is also freely distributable.

The IJG distribution formerly included code to read and write GIF files.
To avoid entanglement with the Unisys LZW patent, GIF reading support has
been removed altogether, and the GIF writer has been simplified to produce
"uncompressed GIFs".  This technique does not use the LZW algorithm; the
resulting GIF files are larger than usual, but are readable by all standard
GIF decoders.

We are required to state that
    "The Graphics Interchange Format(c) is the Copyright property of
    CompuServe Incorporated.  GIF(sm) is a Service Mark property of
    CompuServe Incorporated."

REFERENCES
==========

We recommend reading one or more of these references before trying to
understand the innards of the JPEG software.

The best short technical introduction to the JPEG compression algorithm is
        Wallace, Gregory K.  "The JPEG Still Picture Compression Standard",
        Communications of the ACM, April 1991 (vol. 34 no. 4), pp. 30-44.
(Adjacent articles in that issue discuss MPEG motion picture compression,
applications of JPEG, and related topics.)  If you don't have the CACM issue
handy, a PostScript file containing a revised version of Wallace's article is
available at http://www.ijg.org/files/wallace.ps.gz.  The file (actually
a preprint for an article that appeared in IEEE Trans. Consumer Electronics)
omits the sample images that appeared in CACM, but it includes corrections
and some added material.  Note: the Wallace article is copyright ACM and IEEE,
and it may not be used for commercial purposes.

A somewhat less technical, more leisurely introduction to JPEG can be found in
"The Data Compression Book" by Mark Nelson and Jean-loup Gailly, published by
M&T Books (New York), 2nd ed. 1996, ISBN 1-55851-434-1.  This book provides
good explanations and example C code for a multitude of compression methods
including JPEG.  It is an excellent source if you are comfortable reading C
code but don't know much about data compression in general.  The book's JPEG
sample code is far from industrial-strength, but when you are ready to look
at a full implementation, you've got one here...

The best currently available description of JPEG is the textbook "JPEG Still
Image Data Compression Standard" by William B. Pennebaker and Joan L.
Mitchell, published by Van Nostrand Reinhold, 1993, ISBN 0-442-01272-1.
Price US$59.95, 638 pp.  The book includes the complete text of the ISO JPEG
standards (DIS 10918-1 and draft DIS 10918-2).
Although this is by far the most detailed and comprehensive exposition of
JPEG publicly available, we point out that it is still missing an explanation
of the most essential properties and algorithms of the underlying DCT
technology.
If you think that you know about DCT-based JPEG after reading this book,
then you are in delusion.  The real fundamentals and corresponding potential
of DCT-based JPEG are not publicly known so far, and that is the reason for

all the mistaken developments taking place in the image coding domain.

The original JPEG standard is divided into two parts, Part 1 being the actual specification, while Part 2 covers compliance testing methods. Part 1 is titled "Digital Compression and Coding of Continuous-tone Still Images, Part 1: Requirements and guidelines" and has document numbers ISO/IEC IS 10918-1, ITU-T T.81. Part 2 is titled "Digital Compression and Coding of Continuous-tone Still Images, Part 2: Compliance testing" and has document numbers ISO/IEC IS 10918-2, ITU-T T.83.
IJG JPEG 8 introduces an implementation of the JPEG SmartScale extension which is specified in two documents: A contributed document at ITU and ISO with title "ITU-T JPEG-Plus Proposal for Extending ITU-T T.81 for Advanced Image Coding", April 2006, Geneva, Switzerland. The latest version of this document is Revision 3. And a contributed document ISO/IEC JTC1/SC29/WG1 N 5799 with title "Evolution of JPEG", June/July 2011, Berlin, Germany.

The JPEG standard does not specify all details of an interchangeable file format. For the omitted details we follow the "JFIF" conventions, revision 1.02. JFIF 1.02 has been adopted as an Ecma International Technical Report and thus received a formal publication status. It is available as a free download in PDF format from
http://www.ecma-international.org/publications/techreports/E-TR-098.htm.
A PostScript version of the JFIF document is available at
http://www.ijg.org/files/jfif.ps.gz. There is also a plain text version at
http://www.ijg.org/files/jfif.txt.gz, but it is missing the figures.

The TIFF 6.0 file format specification can be obtained by FTP from
ftp://ftp.sgi.com/graphics/tiff/TIFF6.ps.gz. The JPEG incorporation scheme found in the TIFF 6.0 spec of 3-June-92 has a number of serious problems.
IJG does not recommend use of the TIFF 6.0 design (TIFF Compression tag 6).
Instead, we recommend the JPEG design proposed by TIFF Technical Note #2 (Compression tag 7). Copies of this Note can be obtained from
http://www.ijg.org/files/. It is expected that the next revision
of the TIFF spec will replace the 6.0 JPEG design with the Note's design.
Although IJG's own code does not support TIFF/JPEG, the free libtiff library uses our library to implement TIFF/JPEG per the Note.

## ARCHIVE LOCATIONS

The "official" archive site for this software is www.ijg.org.
The most recent released version can always be found there in
directory "files". This particular version will be archived as
http://www.ijg.org/files/jpegsrc.v8d.tar.gz, and in Windows-compatible "zip" archive format as http://www.ijg.org/files/jpegsr8d.zip.

The JPEG FAQ (Frequently Asked Questions) article is a source of some general information about JPEG.
It is available on the World Wide Web at http://www.faqs.org/faqs/jpeg-faq/ and other news.answers archive sites, including the official news.answers archive at rtfm.mit.edu: ftp://rtfm.mit.edu/pub/usenet/news.answers/jpeg-faq/.
If you don't have Web or FTP access, send e-mail to mail-server@rtfm.mit.edu with body

send usenet/news.answers/jpeg-faq/part1
send usenet/news.answers/jpeg-faq/part2

## ACKNOWLEDGMENTS

Thank to Juergen Bruder for providing me with a copy of the common DCT
algorithm article, only to find out that I had come to the same result
in a more direct and comprehensible way with a more generative approach.

Thank to Istvan Sebestyen and Joan L. Mitchell for inviting me to the
ITU JPEG (Study Group 16) meeting in Geneva, Switzerland.

Thank to Thomas Wiegand and Gary Sullivan for inviting me to the
Joint Video Team (MPEG & ITU) meeting in Geneva, Switzerland.

Thank to Thomas Richter and Daniel Lee for inviting me to the
ISO/IEC JTC1/SC29/WG1 (also known as JPEG, together with ITU-T SG16)
meeting in Berlin, Germany.

Thank to John Korejwa and Massimo Ballerini for inviting me to
fruitful consultations in Boston, MA and Milan, Italy.

Thank to Hendrik Elstner, Roland Fassauer, Simone Zuck, Guenther
Maier-Gerber, Walter Stoeber, Fred Schmitz, and Norbert Braunagel
for corresponding business development.

Thank to Nico Zschach and Dirk Stelling of the technical support team
at the Digital Images company in Halle for providing me with extra
equipment for configuration tests.

Thank to Richard F. Lyon (then of Foveon Inc.) for fruitful
communication about JPEG configuration in Sigma Photo Pro software.

Thank to Andrew Finkenstadt for hosting the ijg.org site.

Last but not least special thank to Thomas G. Lane for the original
design and development of this singular software package.

## FILE FORMAT WARS

The ISO/IEC JTC1/SC29/WG1 standards committee (also known as JPEG, together
with ITU-T SG16) currently promotes different formats containing the name
"JPEG" which is misleading because these formats are incompatible with
original DCT-based JPEG and are based on faulty technologies.
IJG therefore does not and will not support such momentary mistakes
(see REFERENCES).
There exist also distributions under the name "OpenJPEG" promoting such
kind of formats which is misleading because they don't support original
JPEG images.
We have no sympathy for the promotion of inferior formats.  Indeed, one of

the original reasons for developing this free software was to help force convergence on common, interoperable format standards for JPEG files.
Don't use an incompatible file format!
(In any case, our decoder will remain capable of reading existing JPEG image files indefinitely.)

Furthermore, the ISO committee pretends to be "responsible for the popular JPEG" in their public reports which is not true because they don't respond to actual requirements for the maintenance of the original JPEG specification.

There are currently distributions in circulation containing the name "libjpeg" which claim to be a "derivative" or "fork" of the original libjpeg, but don't have the features and are incompatible with formats supported by actual IJG libjpeg distributions. Furthermore, they violate the license conditions as described under LEGAL ISSUES above. We have no sympathy for the release of misleading and illegal distributions derived from obsolete code bases.
Don't use an obsolete code base!

TO DO
=====

Version 8 is the first release of a new generation JPEG standard to overcome the limitations of the original JPEG specification.
More features are being prepared for coming releases...

Please send bug reports, offers of help, etc. to jpeg-info@jpegclub.org.