Open Source Used In JVDI Client for Linux Release 12.7

Cisco Systems, Inc.
www.cisco.com

Cisco has more than 200 offices worldwide.
Addresses, phone numbers, and fax numbers
are listed on the Cisco website at
www.cisco.com/go/offices.

Text Part Number: 78EE117C99-206179157

This document contains licenses and notices for open source software used in this product. With respect to the free/open source software listed in this document, if you have any questions or wish to receive a copy of any source code to which you may be entitled under the applicable free/open source license(s) (such as the GNU Lesser/General Public License), please contact us at external-opensource-requests@cisco.com.

In your requests please include the following reference number 78EE117C99-206179157

Contents

1.1 BOOST C++ Library 1.65.1

1.1.1 Available under license :

This package was debianized by Vladimir Prus <ghost@cs.msu.su> on
Wed, 17 July 2002, 19:27:00 +0400.

Copyright:

  /+\
  +\ Copyright 1993-2002 Christopher Seiwald and Perforce Software, Inc.
  \+/

This is Release 2.4 of Jam/MR, a make-like program.

License is hereby granted to use this software and distribute it
freely, as long as this copyright notice is retained and modifications
are clearly marked.

ALL WARRANTIES ARE HEREBY DISCLAIMED.

Some portions are also:

Copyright 2001-2006 David Abrahams.
Copyright 2002-2006 Rene Rivera.
Copyright 2003-2006 Vladimir Prus.

Distributed under the Boost Software License, Version 1.0.
(See accompanying file LICENSE_1_0.txt or http://www.boost.org/LICENSE_1_0.txt)
Software License, Version 1.0

Copyright 2002-2003, Trustees of Indiana University.
Copyright 2000-2001, University of Notre Dame.
All rights reserved.

Indiana University has the exclusive rights to license this product under the
following license.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

 * All redistributions of source code must retain the above copyright notice,
   the list of authors in the original source code, this list of conditions
   and the disclaimer listed in this license;

 * All redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the disclaimer listed in this license
   in the documentation and/or other materials provided with the distribution;

 * Any documentation included with all redistributions must include the
   following acknowledgement:

   "This product includes software developed at the University of Notre Dame
   and the Pervasive Technology Labs at Indiana University. For technical
   information contact Andrew Lumsdaine at the Pervasive Technology Labs at
   Indiana University. For administrative and license questions contact the
   Advanced Research and Technology Institute at 351 West 10th Street.
   Indianapolis, Indiana 46202, phone 317-278-4100, fax 317-274-5902."

   Alternatively, this acknowledgement may appear in the software itself, and
   wherever such third-party acknowledgments normally appear.

 * The name Indiana University, the University of Notre Dame or "Caramel"
   shall not be used to endorse or promote products derived from this software
   without prior written permission from Indiana University. For written
   permission, please contact Indiana University Advanced Research &
   Technology Institute.

* Products derived from this software may not be called "Caramel", nor may
   Indiana University, the University of Notre Dame or "Caramel" appear in
   their name, without prior written permission of Indiana University Advanced
   Research & Technology Institute.

Indiana University provides no reassurances that the source code provided does
not infringe the patent or any other intellectual property rights of any other
entity. Indiana University disclaims any liability to any recipient for claims
brought by any other entity based on infringement of intellectual property
rights or otherwise.

LICENSEE UNDERSTANDS THAT SOFTWARE IS PROVIDED "AS IS" FOR WHICH NO WARRANTIES
AS TO CAPABILITIES OR ACCURACY ARE MADE. INDIANA UNIVERSITY GIVES NO WARRANTIES
AND MAKES NO REPRESENTATION THAT SOFTWARE IS FREE OF INFRINGEMENT OF THIRD
PARTY PATENT, COPYRIGHT, OR OTHER PROPRIETARY RIGHTS. INDIANA UNIVERSITY MAKES
NO WARRANTIES THAT SOFTWARE IS FREE FROM "BUGS", "VIRUSES", "TROJAN HORSES",
"TRAP DOORS", "WORMS", OR OTHER HARMFUL CODE. LICENSEE ASSUMES THE ENTIRE RISK
AS TO THE PERFORMANCE OF SOFTWARE AND/OR ASSOCIATED MATERIALS, AND TO THE
PERFORMANCE AND VALIDITY OF INFORMATION GENERATED USING SOFTWARE.
The following people hereby grant permission to replace all existing
licenses on their contributions to Boost with the Boost Software
License, Version 1.0. (boostinspect:nolicense boostinspect:nocopyright)

Aleksey Gurtovoy (agurtovoy@meta-comm.com)
Andrei Alexandrescu (andrewalex - at - hotmail.com) (See Boost list message of August 12, 2004 11:06:58 AM EST)
Andrew Lumsdaine ()
Anthony Williams (anthony -at- justsoftwaresolutions.co.uk)
Beman Dawes (bdawes@acm.org)
Brad King (brad.king -at- kitware.com) (See Boost list message of Wed, 21 Jul 2004 11:15:46 -0400)
Brian Osman (osman -at- vvisions.com) (See CVS log)
Bruce Barr (schmoost -at- yahoo.com) (See Boost list of Mon, 16 Aug 2004 15:06:43 -0500)
Bruno da Silva de Oliveira (bruno - at - esss.com.br)
Christain Engstrom (christian.engstrom -at- glindra.org) (See Boost list message of Mon, 30 Aug 2004 14:31:49 +0200)
Cromwell D Enage (sponage -at- yahoo.com) (See Boost list message of August 12, 2004 11:49:13 AM EST)
Dan Gohman (djg -at- cray.com) (See Boost list messsage of Sat, 21 Aug 2004 10:54:59 +0100)
Dan Nuffer (dan -at- nuffer.name)
Daniel Frey (d.frey -at- gmx.de, daniel.frey -at- aixigo.de)
Daniel Nuffer (dan -at- nuffer.name)
Darin Adler (darin -at- bentspoon.com) (Email to Andreas Huber, see change log)
Daryle Walker (darylew - at - hotmail.com)
Dave Abrahams (dave@boost-consulting.com)
Dave Moore (dmoore -at- viefinancial.com) (See Boost list message of 18 Dec 2003 15:35:50 -0500)
David Abrahams (dave@boost-consulting.com)
Dietmar Kuehl (dietmar_kuehl -at- yahoo.com) (Email to Andreas Huber, see change log)
Douglas Gregor (gregod -at- cs.rpi.edu, dgregor -at- cs.indiana.edu, doug.gregor -at- gmail.com)
Dr John Maddock (john - at - johnmaddock.co.uk)

Edward D. Brey (brey -at- ductape.net) (Email to Andreas Huber, see change log)

Eric Ford (un5o6n902 -at- sneakemail.com) (See Boost list message of Sun, 15 Aug 2004 10:29:13 +0100)

Eric Friedman (ebf@users.sourceforge.net)

Eric Niebler (eric@boost-consulting.com)

Fernando Cacciola (fernando_cacciola@ciudad.com.ar)

Fernando Luis Cacciola Carballal (fernando_cacciola@ciudad.com.ar)

Francois Faure (Francois.Faure -at- imag.fr) (See CVS log)

Gary Powell (powellg - at - amazon.com) (See Boost list message of 10 Feb 2004 14:22:46 -0800)

Gennadiy Rozental (rogeeff -at- mail.com) (Email to Andreas Huber, see change log)

Gottfried Ganssauge (Gottfried.Ganssauge -at- HAUFE.DE) (See Boost List message of Mon, 16 Aug 2004 10:09:19 +0200)

Gottfried Ganauge (Gottfried.Ganssauge -at- HAUFE.DE) (Alternative spelling of Gottfried Ganssauge)

Greg Colvin (gregory.colvin -at- oracle.com) (See Boost list message of Sat, 14 Aug 2004 10:57:00 +0100)

Gregory Colvin (gregory.colvin -at- oracle.com) (See Boost list message of Sat, 14 Aug 2004 10:57:00 +0100)

Gunter Winkler (gunter.winkler -at- unibw-muenchen.de) (See Boost List message of Mon, 16 Aug 2004 10:24:17 +0200)

Hartmut Kaiser (hartmut.kaiser -at- gmail.com)

Herve Bronnimann (hbr -at- poly.edu)

Herv Brnnimann (hbr -at- poly.edu)

Housemarque Oy (Ilari Kuittinen ilari.kuittinen -at- housemarque.fi)

Howard Hinnant (hinnant -at- twcny.rr.com) (See Boost list message of July 25, 2004 3:44:49 PM EST)

Hubert Holin (hubert_holin -at- users.sourceforge.net)

Indiana University ()

Itay Maman (imaman -at- users.sourceforge.net)

Jaakko Jrvi (jajarvi -at- osl.iu.edu)

Jaap Suter (j.suter -at- student.utwente.nl) (See Boost list message of Thu, 16 Sep 2004 09:32:43 -0700)

Jeff Garland (jeff - at - crystalclearsoftware.com) (see Boost list post of July 25, 2004 19:31:09 -0700)

Jens Maurer (Jens.Maurer@gmx.net)

Jeremy G Siek (jsiek@osl.iu.edu)

Jeremy Siek (jsiek@osl.iu.edu)

Joel de Guzman (joel -at- boost-consulting.com) (See Boost list message of July 25, 2004 8:32:00 PM EST)

John Bandela (jbandela-at-ufl.edu)

John Maddock (john - at - johnmaddock.co.uk)

John R Bandela (jbandela-at-ufl.edu)

Jonathan Turkanis (turkanis -at- coderage dot com)

Juergen Hunold (hunold -at- ive.uni-hannover.de) (See Boost List Message of Fri, 13 Aug 2004 19:39:55 +0200)

Kevlin Henney (kevlin -at- curbralan.com) (See Boost list message of Wed, 15 Sep 2004 18:15:17 +0200)

Kresimir Fresl (fresl -at- master.grad.hr) (See Boost List message of August 16, 2004 8:23:35 AM EST)

Lars Gullik Bjnnes (larsbj -at- lyx.org) (See Boost list message of Tue, 17 Aug 2004 15:49:02 +0100)

Lie-Quan Lee (liequan - at - slac.stanford.edu, llee - at - cs.indiana.edu)

Maarten Keijzer (mkeijzer -at- cs.vu.nl) (See Boost list message of Wed, 18 Aug 2004 21:43:18 +0100)

Mac Murrett (mmurrett -at- mac.com)

Marc Wintermantel (wintermantel -at- imes.mavt.ethz.ch, wintermantel -at- even-ag.ch) (See CVS log)

Michael Glassford (glassfordm - at - hotmail.com)

Michael Stevens (Michael.Stevens - at - epost.de)

Multi Media Ltd. (pdimov@mmltd.net)

Nicolai M Josuttis (solutions -at- josuttis.com) (See Boost list message of Mon, 30 Aug 2004 10:52:00 +0100)

Nikolay Mladenov (nickm -at- sitius.com) (See Boost list message of Tue, 17 Aug 2004 15:45:33 +0100)

Paul Mensonides (pmenso57 -at- comcast.net) (See Boost list message of July 21, 2004 1:12:21 AM EST)

Pavol Droba (droba -at- topmail.sk)

Peter Dimov (pdimov@mmltd.net)

R W Grosse-Kunstleve (RWGrosse-Kunstleve@lbl.gov)

Ralf W. Grosse-Kunstleve (RWGrosse-Kunstleve@lbl.gov)

Rational Discovery LLC (Greg Landrum Landrum -at- RationalDiscovery.com) (See Boost list post of Tue, 17 Aug 2004 10:35:36 +0100)

Rene Rivera (grafik/redshift-software.com, rrivera/acm.org)

Robert Ramey (ramey@www.rrsd.com)

Roland Richter (roland -at- flll.jku.at) (See Boost list post of Mon, 16 Aug 2004 22:16:55 +0200)

Roland Schwarz (roland.schwarz -at- chello.at)

Ronald Garcia (garcia -at- cs.indiana.edu) (Email to Andreas Huber, see change log)

Samuel Krempp (krempp -at- crans.ens-cachan.fr) (See Boost list message of Mon, 27 Sep 2004 13:18:36 +0200)

Stefan Seefeld (seefeld -at- sympatico.ca)

Stephen Cleary (scleary -at- jerviswebb.com) (See Boost list message of Tue, 28 Sep 2004 13:11:46 +0100)

Steve Cleary (Variant of Stephen Cleary)

Sylvain Pion (Sylvain.Pion - at - sophia.inria.fr)

The Trustees of Indiana University ()

Thomas Witt (witt - at - ive.uni-hannover.de, witt - at - acm.org, witt - at - styleadvisor.com)

Thorsten Jrgen Ottosen (nesotto - at - cs.auc.dk)

Thorsten Ottosen (nesotto - at - cs.auc.dk)

Toon Knapen (toon dot knapen - at - fft.be)

Trustees of Indiana University ()

University of Notre Dame ()

Vladimir Prus (ghost@cs.msu.su)

William E. Kempf () (email to Beman Dawes, 9/14/2006 4:18 PM)

Joerg Walter (jhr.walter - at - t-online.de : email to ublas mailing list Mon, 17 Sep 2007 10:17:08 +0200)

Mathias Koch (mkoch - at - idesis.de 7 : email to boost-owner@lists.boost.org Sep 2007 13:20:09 +0200)

--- end ---

Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization
obtaining a copy of the software and accompanying documentation covered by
this license (the "Software") to use, reproduce, display, distribute,
execute, and transmit the Software, and to prepare derivative works of the
Software, and to permit third-parties to whom the Software is furnished to
do so, all subject to the following:

The copyright notices in the Software and this entire statement, including
the above license grant, this restriction and the following disclaimer,
must be included in all copies of the Software, in whole or in part, and
all derivative works of the Software, unless such copies or derivative
works are solely in the form of machine-executable object code generated by
a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,

FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE article PUBLIC "-//Boost//DTD BoostBook XML V1.0//EN"
"http://www.boost.org/tools/boostbook/dtd/boostbook.dtd">
<article id="copyright_test" last-revision="DEBUG MODE Date: 2000/12/20 12:00:00 $"
xmlns:xi="http://www.w3.org/2001/XInclude">
 <title>Copyright Test</title>
 <articleinfo>
  <copyright>
   <year>1963</year> <year>1964</year> <year>1965</year> <holder>Jane Doe</holder>
  </copyright>
  <copyright>
   <year>2018</year> <holder>Joe Blow, John Coe</holder>
  </copyright>
  <copyright>
   <year>1977</year> <year>1985</year> <holder>Someone else</holder>
  </copyright>
 </articleinfo>
</article>
[article Copyright Test
[quickbook 1.5]
[copyright 1963-1965 Jane Doe, 2018 Joe Blow, John Coe]
[copyright 1977,1985 Someone else]
]
[article Invalid copyright
[quickbook 1.5]
[copyright No year]
]

Maybe this should pass?
[article Invalid copyright years
[quickbook 1.5]
[copyright 2010-2001 Timequake man]
]
/*
 *
 * Copyright (c) 2003 Dr John Maddock
 * Use, modification and distribution is subject to the
 * Boost Software License, Version 1.0. (See accompanying file
 * LICENSE_1_0.txt or copy at http://www.boost.org/LICENSE_1_0.txt)
 *
 */

#include "licence_info.hpp"
```

```cpp
#include "bcp_imp.hpp"
#include "fileview.hpp"
#include <fstream>
#include <iomanip>
#include <cstring>
#include <stdexcept>
#include <boost/lexical_cast.hpp>
#include <boost/filesystem/operations.hpp>
#include <boost/throw_exception.hpp>

//
// split_path is a small helper for outputting a path name,
// complete with a link to that path:
//
struct split_path
{
  const fs::path& root;
  const fs::path& file;
  split_path(const fs::path& r, const fs::path& f)
    : root(r), file(f){}
private:
  split_path& operator=(const split_path&);
};

std::ostream& operator << (std::ostream& os, const split_path& p)
{
  os << "<a href=\"" << (p.root / p.file).string() << "\">" << p.file.string() << "</a>";
  return os;
}

std::string make_link_target(const std::string& s)
{
  // convert an arbitrary string into something suitable
  // for an <a> name:
  std::string result;
  for(unsigned i = 0; i < s.size(); ++i)
  {
    result.append(1, static_cast<std::string::value_type>(std::isalnum(s[i]) ? s[i] : '_'));
  }
  return result;
}


void bcp_implementation::output_license_info()
{
  std::pair<const license_info*, int> licenses = get_licenses();

  std::map<int, license_data>::const_iterator i, j;
```

```cpp
i = m_license_data.begin();
j = m_license_data.end();

std::ofstream os(m_dest_path.string().c_str());
if(!os)
{
   std::string msg("Error opening ");
   msg += m_dest_path.string();
   msg += " for output.";
   std::runtime_error e(msg);
   boost::throw_exception(e);
}
os <<
   "<!DOCTYPE HTML PUBLIC \"-//W3C//DTD HTML 4.0 Transitional//EN\">\n"
   "<html>\n"
   "<head>\n"
   "<title>Boost Licence Dependency Information";
if(m_module_list.size() == 1)
{
   os << " for " << *(m_module_list.begin());
}
os <<
   "</title>\n"
   "</head>\n"
   "<body>\n"
   "<H1>Boost Licence Dependency Information";
if(m_module_list.size() == 1)
{
   os << " for " << *(m_module_list.begin());
}
os <<
   "</H1>\n"
   "<H2>Contents</h2>\n"
   "<pre><a href=\"#input\">Input Information</a>\n";
if(!m_bsl_summary_mode)
   os << "<a href=\"#summary\">Licence Summary</a>\n";
os << "<a href=\"#details\">Licence Details</a>\n";

while(i != j)
{
   // title:
   os << "   <A href=\"#" << make_link_target(licenses.first[i->first].license_name)
      << "\">" << licenses.first[i->first].license_name << "</a>\n";
   ++i;
}

os << "<a href=\"#files\">Files with no recognised license</a>\n"
   "<a href=\"#authors\">Files with no recognised copyright holder</a>\n";
```

```cpp
if(!m_bsl_summary_mode)
{
  os <<
  "Moving to the Boost Software License...\n"
  "  <a href=\"#bsl-converted\">Files that can be automatically converted to the Boost Software License</a>\n"
  "  <a href=\"#to-bsl\">Files that can be manually converted to the Boost Software License</a>\n"
  "  <a href=\"#not-to-bsl\">Files that can <b>NOT</b> be moved to the Boost Software License</a>\n"
  "  <a href=\"#need-bsl-authors\">Authors we need to move to the Boost Software License</a>\n"
  "<a href=\"#copyright\">Copyright Holder Information</a>\n";
}
os <<
  "<a href=\"#depend\">File Dependency Information</a>\n"
  "</pre>";

//
// input Information:
//
os << "<a name=\"input\"></a><h2>Input Information</h2>\n";
if(m_scan_mode)
  os << "<P>The following files were scanned for boost dependencies:<BR>";
else
  os << "<P>The following Boost modules were checked:<BR>";

std::list<std::string>::const_iterator si = m_module_list.begin();
std::list<std::string>::const_iterator sj = m_module_list.end();
while(si != sj)
{
  os << *si << "<BR>";
  ++si;
}
os << "</p><p>The Boost path was: <code>" << m_boost_path.string() << "</code></P>";
//
// extract the boost version number from the boost directory tree,
// not from this app (which may have been built from a previous
// version):
//
fileview version_file(m_boost_path / "boost/version.hpp");
static const boost::regex version_regex(
  "^[[:blank:]]*#[[:blank:]]*define[[:blank:]]+BOOST_VERSION[[:blank:]]+(\\d+)");
boost::cmatch what;
if(boost::regex_search(version_file.begin(), version_file.end(), what, version_regex))
{
  int version = boost::lexical_cast<int>(what.str(1));
  os << "<p>The Boost version is: " << version / 100000 << "." << version / 100 % 1000 << "." << version % 100
<< "</P>\n";
}

//
```

```cpp
// output each license:
//
i = m_license_data.begin();
j = m_license_data.end();
if(!m_bsl_summary_mode)
{
  //
  // start with the summary:
  //
  os << "<a name=\"summary\"></a><h2>Licence Summary</h2>\n";
  while(i != j)
  {
    // title:
    os <<
      "<H3>" << licenses.first[i->first].license_name << "</H3>\n";
    // license text:
    os << "<BLOCKQUOTE>" << licenses.first[i->first].license_text << "</BLOCKQUOTE>";
    // Copyright holders:
    os << "<P>This license is used by " << i->second.authors.size()
      << " authors and " << i->second.files.size()
      << " files <a href=\"#" << make_link_target(licenses.first[i->first].license_name) << "\">(see details)</a>";
    os << "</P></BLOCKQUOTE>\n";
    ++i;
  }
}
//
// and now the details:
//
i = m_license_data.begin();
j = m_license_data.end();
int license_index = 0;
os << "<a name=\"details\"></a><h2>Licence Details</h2>\n";
while(i != j)
{
  // title:
  os <<
    "<H3><A name=\"" << make_link_target(licenses.first[i->first].license_name)
    << "\"></a>" << licenses.first[i->first].license_name << "</H3>\n";
  // license text:
  os << "<BLOCKQUOTE>" << licenses.first[i->first].license_text << "</BLOCKQUOTE>";
  if(!m_bsl_summary_mode || (license_index >= 3))
  {
    // Copyright holders:
    os << "<P>This license is used by the following " << i->second.authors.size() << " copyright
holders:</P>\n<BLOCKQUOTE><P>";
    std::set<std::string>::const_iterator x, y;
    x = i->second.authors.begin();
    y = i->second.authors.end();
```

```cpp
        while(x != y)
        {
          os << *x << "<BR>\n";
          ++x;
        }
        os << "</P></BLOCKQUOTE>\n";
        // Files using this license:
        os << "<P>This license applies to the following " << i->second.files.size() << "
files:</P>\n<BLOCKQUOTE><P>";
        std::set<fs::path, path_less>::const_iterator m, n;
        m = i->second.files.begin();
        n = i->second.files.end();
        while(m != n)
        {
          os << split_path(m_boost_path, *m) << "<br>\n";
          ++m;
        }
        os << "</P></BLOCKQUOTE>\n";
      }
      else
      {
        os << "<P>This license is used by " << i->second.authors.size() << " authors (list omitted for brevity).</P>\n";
        os << "<P>This license applies to " << i->second.files.size() << " files (list omitted for brevity).</P>\n";
      }
      ++license_index;
      ++i;
   }
   //
   // Output list of files not found to be under license control:
   //
   os << "<h2><a name=\"files\"></a>Files With No Recognisable Licence</h2>\n"
      "<P>The following " << m_unknown_licenses.size() << " files had no recognisable license
information:</P><BLOCKQUOTE><P>\n";
   std::set<fs::path, path_less>::const_iterator i2, j2;
   i2 = m_unknown_licenses.begin();
   j2 = m_unknown_licenses.end();
   while(i2 != j2)
   {
     os << split_path(m_boost_path, *i2) << "<br>\n";
     ++i2;
   }
   os << "</p></BLOCKQUOTE>";
   //
   // Output list of files with no found copyright holder:
   //
   os << "<h2><a name=\"authors\"></a>Files With No Recognisable Copyright Holder</h2>\n"
      "<P>The following " << m_unknown_authors.size() << " files had no recognisable copyright
holder:</P>\n<BLOCKQUOTE><P>";
```

```cpp
  i2 = m_unknown_authors.begin();
  j2 = m_unknown_authors.end();
  while(i2 != j2)
  {
    os << split_path(m_boost_path, *i2) << "<br>\n";
    ++i2;
  }
  os << "</p></BLOCKQUOTE>";
  if(!m_bsl_summary_mode)
  {
    //
    // Output list of files that have been moved over to the Boost
    // Software License, along with enough information for human
    // verification.
    //
    os << "<h2><a name=\"bsl-converted\"></a>Files that can be automatically converted to the Boost Software
License</h2>\n"
       << "<P>The following " << m_converted_to_bsl.size() << " files can be automatically converted to the Boost
Software License, but require manual verification before they can be committed to CVS:</P>\n";
    if (!m_converted_to_bsl.empty())
    {
      typedef std::map<fs::path, std::pair<std::string, std::string>, path_less>
        ::const_iterator conv_iterator;
      conv_iterator i = m_converted_to_bsl.begin(),
              ie = m_converted_to_bsl.end();
      int file_num = 1;
      while (i != ie)
      {
        os << "<P>[" << file_num << "] File: <tt>" << split_path(m_boost_path, i->first)
          << "</tt><br>\n<table border=\"1\">\n  <tr>\n    <td><pre>"
          << i->second.first << "</pre></td>\n    <td><pre>"
          << i->second.second << "</pre></td>\n  </tr>\n</table>\n";
        ++i;
        ++file_num;
      }
    }
    //
    // Output list of files that could be moved over to the Boost Software License
    //
    os << "<h2><a name=\"to-bsl\"></a>Files that could be converted to the Boost Software License</h2>\n"
      "<P>The following " << m_can_migrate_to_bsl.size() << " files could be manually converted to the Boost
Software License, but have not yet been:</P>\n<BLOCKQUOTE><P>";
    i2 = m_can_migrate_to_bsl.begin();
    j2 = m_can_migrate_to_bsl.end();
    while(i2 != j2)
    {
      os << split_path(m_boost_path, *i2) << "<br>\n";
      ++i2;
```

```cpp
   }
   os << "</p></BLOCKQUOTE>";
   //
   // Output list of files that can not be moved over to the Boost Software License
   //
   os << "<h2><a name=\"not-to-bsl\"></a>Files that can NOT be converted to the Boost Software
License</h2>\n"
      "<P>The following " << m_cannot_migrate_to_bsl.size() << " files cannot be converted to the Boost Software
License because we need the permission of more authors:</P>\n<BLOCKQUOTE><P>";
   i2 = m_cannot_migrate_to_bsl.begin();
   j2 = m_cannot_migrate_to_bsl.end();
   while(i2 != j2)
   {
      os << split_path(m_boost_path, *i2) << "<br>\n";
      ++i2;
   }
   os << "</p></BLOCKQUOTE>";
   //
   // Output list of authors that we need permission for to move to the BSL
   //
   os << "<h2><a name=\"need-bsl-authors\"></a>Authors we need for the BSL</h2>\n"
      "<P>Permission of the following authors is needed before we can convert to the Boost Software License. The
list of authors that have given their permission is contained in <code>more/blanket-
permission.txt</code>.</P>\n<BLOCKQUOTE><P>";
   std::copy(m_authors_for_bsl_migration.begin(), m_authors_for_bsl_migration.end(),
         std::ostream_iterator<std::string>(os, "<br>\n"));
   os << "</p></BLOCKQUOTE>";
   //
   // output a table of copyright information:
   //
   os << "<H2><a name=\"copyright\"></a>Copyright Holder Information</H2><table border=\"1\">\n";
   std::map<std::string, std::set<fs::path, path_less> >::const_iterator ad, ead;
   ad = m_author_data.begin();
   ead = m_author_data.end();
   while(ad != ead)
   {
      os << "<tr><td>" << ad->first << "</td><td>";
      std::set<fs::path, path_less>::const_iterator fi, efi;
      fi = ad->second.begin();
      efi = ad->second.end();
      while(fi != efi)
      {
         os << split_path(m_boost_path, *fi) << " ";
         ++fi;
      }
      os << "</td></tr>\n";
      ++ad;
   }
```

```
      os << "</table>\n";
   }


   //
   // output file dependency information:
   //
   os << "<H2><a name=\"depend\"></a>File Dependency Information</H2><BLOCKQUOTE><pre>\n";
   std::map<fs::path, fs::path, path_less>::const_iterator dep, last_dep;
   std::set<fs::path, path_less>::const_iterator fi, efi;
   fi = m_copy_paths.begin();
   efi = m_copy_paths.end();
   // if in summary mode, just figure out the "bad" files and print those only:
   std::set<fs::path, path_less> bad_paths;
   if(m_bsl_summary_mode)
   {
      bad_paths.insert(m_unknown_licenses.begin(), m_unknown_licenses.end());
      bad_paths.insert(m_unknown_authors.begin(), m_unknown_authors.end());
      bad_paths.insert(m_can_migrate_to_bsl.begin(), m_can_migrate_to_bsl.end());
      bad_paths.insert(m_cannot_migrate_to_bsl.begin(), m_cannot_migrate_to_bsl.end());
      typedef std::map<fs::path, std::pair<std::string, std::string>, path_less>
         ::const_iterator conv_iterator;
      conv_iterator i = m_converted_to_bsl.begin(),
              ie = m_converted_to_bsl.end();
      while(i != ie)
      {
         bad_paths.insert(i->first);
         ++i;
      }
      fi = bad_paths.begin();
      efi = bad_paths.end();
      os << "<P>For brevity, only files not under the BSL are shown</P>\n";
   }
   while(fi != efi)
   {
      os << split_path(m_boost_path, *fi);
      dep = m_dependencies.find(*fi);
      last_dep = m_dependencies.end();
      std::set<fs::path, path_less> seen_deps;
      if (dep != last_dep)
        while(true)
          {
           os << " -> ";
           if(fs::exists(m_boost_path / dep->second))
             os << split_path(m_boost_path, dep->second);
           else if(fs::exists(dep->second))
             os << split_path(fs::path(), dep->second);
           else
             os << dep->second.string();
```

```
          if(seen_deps.find(dep->second) != seen_deps.end())
            {
             os << " <I>(Circular dependency!)</I>";
             break; // circular dependency!!!
            }
          seen_deps.insert(dep->second);
          last_dep = dep;
          dep = m_dependencies.find(dep->second);
          if((dep == m_dependencies.end()) || (0 == compare_paths(dep->second, last_dep->second)))
            break;
        }
    os << "\n";
    ++fi;
  }
  os << "</pre></BLOCKQUOTE>\n";

  os << "</body></html>\n";

  if(!os)
  {
    std::string msg("Error writing to ");
    msg += m_dest_path.string();
    msg += ".";
    std::runtime_error e(msg);
    boost::throw_exception(e);
  }

}
```
Software License, Version 1.0

Indiana University has the exclusive rights to license this product under the
following license.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

 * All redistributions of source code must retain the above copyright notice,
   the list of authors in the original source code, this list of conditions
   and the disclaimer listed in this license;

 * All redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the disclaimer listed in this license
   in the documentation and/or other materials provided with the distribution;

* Any documentation included with all redistributions must include the
  following acknowledgement:

  "This product includes software developed at the University of Notre Dame
  and the Pervasive Technology Labs at Indiana University. For technical
  information contact Andrew Lumsdaine at the Pervasive Technology Labs at
  Indiana University. For administrative and license questions contact the
  Advanced Research and Technology Institute at 351 West 10th Street.
  Indianapolis, Indiana 46202, phone 317-278-4100, fax 317-274-5902."

  Alternatively, this acknowledgement may appear in the software itself, and
  wherever such third-party acknowledgments normally appear.

* The name Indiana University, the University of Notre Dame or "Caramel"
  shall not be used to endorse or promote products derived from this software
  without prior written permission from Indiana University. For written
  permission, please contact Indiana University Advanced Research &
  Technology Institute.

* Products derived from this software may not be called "Caramel", nor may
  Indiana University, the University of Notre Dame or "Caramel" appear in
  their name, without prior written permission of Indiana University Advanced
  Research & Technology Institute.

Indiana University provides no reassurances that the source code provided does
not infringe the patent or any other intellectual property rights of any other
entity. Indiana University disclaims any liability to any recipient for claims
brought by any other entity based on infringement of intellectual property
rights or otherwise.

LICENSEE UNDERSTANDS THAT SOFTWARE IS PROVIDED "AS IS" FOR WHICH NO WARRANTIES
AS TO CAPABILITIES OR ACCURACY ARE MADE. INDIANA UNIVERSITY GIVES NO WARRANTIES
AND MAKES NO REPRESENTATION THAT SOFTWARE IS FREE OF INFRINGEMENT OF THIRD
PARTY PATENT, COPYRIGHT, OR OTHER PROPRIETARY RIGHTS. INDIANA UNIVERSITY MAKES
NO WARRANTIES THAT SOFTWARE IS FREE FROM "BUGS", "VIRUSES", "TROJAN HORSES",
"TRAP DOORS", "WORMS", OR OTHER HARMFUL CODE. LICENSEE ASSUMES THE ENTIRE RISK
AS TO THE PERFORMANCE OF SOFTWARE AND/OR ASSOCIATED MATERIALS, AND TO THE
PERFORMANCE AND VALIDITY OF INFORMATION GENERATED USING SOFTWARE.
This package was debianized by Vladimir Prus <ghost@cs.msu.su> on
Wed, 17 July 2002, 19:27:00 +0400.

Copyright:

  /+\
  +\ Copyright 1993-2002 Christopher Seiwald and Perforce Software, Inc.
  \+/

  This is Release 2.4 of Jam/MR, a make-like program.

License is hereby granted to use this software and distribute it
freely, as long as this copyright notice is retained and modifications
are clearly marked.

ALL WARRANTIES ARE HEREBY DISCLAIMED.

Some portions are also:

Copyright 2001-2006 David Abrahams.
Copyright 2002-2006 Rene Rivera.
Copyright 2003-2006 Vladimir Prus.

Distributed under the Boost Software License, Version 1.0.
(See accompanying file LICENSE_1_0.txt or http://www.boost.org/LICENSE_1_0.txt)
This is an attempt to acknowledge early contributions to the garbage
collector. Later contributions should instead be mentioned in
README.changes.

HISTORY -

 Early versions of this collector were developed as a part of research
projects supported in part by the National Science Foundation
and the Defense Advance Research Projects Agency.

The garbage collector originated as part of the run-time system for
the Russell programming language implementation. The first version of the
garbage collector was written primarily by Al Demers. It was then refined
and mostly rewritten, primarily by Hans-J. Boehm, at Cornell U.,
the University of Washington, Rice University (where it was first used for
C and assembly code), Xerox PARC, SGI, and HP Labs. However, significant
contributions have also been made by many others.

Some other contributors:

More recent contributors are mentioned in the modification history in
README.changes. My apologies for any omissions.

The SPARC specific code was originally contributed by Mark Weiser.
The Encore Multimax modifications were supplied by
Kevin Kenny (kenny@m.cs.uiuc.edu). The adaptation to the IBM PC/RT is largely
due to Vernon Lee, on machines made available to Rice by IBM.
Much of the HP specific code and a number of good suggestions for improving the
generic code are due to Walter Underwood.
Robert Brazile (brazile@diamond.bbn.com) originally supplied the ULTRIX code.
Al Dosser (dosser@src.dec.com) and Regis Cridlig (Regis.Cridlig@cl.cam.ac.uk)
subsequently provided updates and information on variation between ULTRIX
systems. Parag Patel (parag@netcom.com) supplied the A/UX code.

Jesper Peterson(jep@mtiame.mtia.oz.au), Michel Schinz, and
Martin Tauchmann (martintauchmann@bigfoot.com) supplied the Amiga port.
Thomas Funke (thf@zelator.in-berlin.de(?)) and
Brian D.Carlstrom (bdc@clark.lcs.mit.edu) supplied the NeXT ports.
Douglas Steel (doug@wg.icl.co.uk) provided ICL DRS6000 code.
Bill Janssen (janssen@parc.xerox.com) supplied the SunOS dynamic loader
specific code. Manuel Serrano (serrano@cornas.inria.fr) supplied linux and
Sony News specific code. Al Dosser provided Alpha/OSF/1 code. He and
Dave Detlefs(detlefs@src.dec.com) also provided several generic bug fixes.
Alistair G. Crooks(agc@uts.amdahl.com) supplied the NetBSD and 386BSD ports.
Jeffrey Hsu (hsu@soda.berkeley.edu) provided the FreeBSD port.
Brent Benson (brent@jade.ssd.csd.harris.com) ported the collector to
a Motorola 88K processor running CX/UX (Harris NightHawk).
Ari Huttunen (Ari.Huttunen@hut.fi) generalized the OS/2 port to
nonIBM development environments (a nontrivial task).
Patrick Beard (beard@cs.ucdavis.edu) provided the initial MacOS port.
David Chase, then at Olivetti Research, suggested several improvements.
Scott Schwartz (schwartz@groucho.cse.psu.edu) supplied some of the
code to save and print call stacks for leak detection on a SPARC.
Jesse Hull and John Ellis supplied the C++ interface code.
Zhong Shao performed much of the experimentation that led to the
current typed allocation facility. (His dynamic type inference code hasn't
made it into the released version of the collector, yet.)
// Copyright Louis Dionne 2013-2017
// Distributed under the Boost Software License, Version 1.0.
// (See accompanying file LICENSE.md or copy at http://boost.org/LICENSE_1_0.txt)

#include <boost/mpl/push_back.hpp>
#include <boost/mpl/quote.hpp>
#include <boost/mpl/transform.hpp>
#include <boost/mpl/vector.hpp>

template <typename X>
struct f { using type = X; };

template <int i>
struct t { };

using vector = <%= mpl_vector((1..input_size).to_a.map { |n| "t<#{n}>" }) %>;

using result = boost::mpl::transform<vector, boost::mpl::quote1<f>>::type;

int main() { }
// Copyright Louis Dionne 2013-2017
// Distributed under the Boost Software License, Version 1.0.
// (See accompanying file LICENSE.md or copy at http://boost.org/LICENSE_1_0.txt)

```cpp
#include <boost/mpl/push_back.hpp>
#include <boost/mpl/vector.hpp>


template <int i>
struct t { };

using vector = <%= mpl_vector((1..input_size).to_a.map { |n| "t<#{n}>" }) %>;

int main() { }
// Copyright Louis Dionne 2013-2017
// Distributed under the Boost Software License, Version 1.0.
// (See accompanying file LICENSE.md or copy at http://boost.org/LICENSE_1_0.txt)

#include <boost/mpl/fold.hpp>
#include <boost/mpl/push_back.hpp>
#include <boost/mpl/quote.hpp>
#include <boost/mpl/vector.hpp>


template <typename State, typename X>
struct f { using type = X; };

struct state { };

template <int i>
struct t { };

using vector = <%= mpl_vector((1..input_size).to_a.map { |n| "t<#{n}>" }) %>;

using result = boost::mpl::fold<vector, state, boost::mpl::quote2<f>>::type;


int main() { }
Copyright Louis Dionne 2013-2017
```

Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization
obtaining a copy of the software and accompanying documentation covered by
this license (the "Software") to use, reproduce, display, distribute,
execute, and transmit the Software, and to prepare derivative works of the
Software, and to permit third-parties to whom the Software is furnished to
do so, all subject to the following:

The copyright notices in the Software and this entire statement, including
the above license grant, this restriction and the following disclaimer,

must be included in all copies of the Software, in whole or in part, and
all derivative works of the Software, unless such copies or derivative
works are solely in the form of machine-executable object code generated by
a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT
SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE
FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.

```
<!--
Copyright Louis Dionne 2013-2017
Distributed under the Boost Software License, Version 1.0.
(See accompanying file LICENSE.md or copy at http://boost.org/LICENSE_1_0.txt)
-->
<!-- boost-no-inspect -->
<!-- HTML header for doxygen 1.8.9.1-->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/xhtml;charset=UTF-8"/>
<meta http-equiv="X-UA-Compatible" content="IE=9"/>
<meta name="generator" content="Doxygen 1.8.11"/>
<title>Boost.Hana: Boost.MPL adapters</title>
<link href="tabs.css" rel="stylesheet" type="text/css"/>
<script type="text/javascript" src="jquery.js"></script>
<script type="text/javascript" src="dynsections.js"></script>
<link href="navtree.css" rel="stylesheet" type="text/css"/>
<script type="text/javascript" src="resize.js"></script>
<script type="text/javascript" src="navtreedata.js"></script>
<script type="text/javascript" src="navtree.js"></script>
<script type="text/javascript">
 $(document).ready(initResizable);
 $(window).load(resizeHeight);
</script>
<link href="search/search.css" rel="stylesheet" type="text/css"/>
<script type="text/javascript" src="search/searchdata.js"></script>
<script type="text/javascript" src="search/search.js"></script>
<script type="text/javascript">
 $(document).ready(function() { init_search(); });
</script>
<script type="text/x-mathjax-config">
 MathJax.Hub.Config({
   extensions: ["tex2jax.js"],
   jax: ["input/TeX","output/HTML-CSS"],
```

```
});
// Copyright Louis Dionne 2013-2017
// Distributed under the Boost Software License, Version 1.0.
// (See accompanying file LICENSE.md or copy at http://boost.org/LICENSE_1_0.txt)
MathJax.Hub.Config({
  "HTML-CSS": {
    linebreaks: {
      automatic: true,
      width: "75% container"
    }
  }
});
</script><script type="text/javascript" src="https://cdn.mathjax.org/mathjax/latest/MathJax.js"></script>
<link href="doxygen.css" rel="stylesheet" type="text/css" />
<!-- Additional javascript for drawing charts. -->
<script type="text/javascript" src="highcharts.js"></script>
<script type="text/javascript" src="highcharts-data.js"></script>
<script type="text/javascript" src="highcharts-exporting.js"></script>
<script type="text/javascript" src="chart.js"></script>
<script type="text/javascript" src="hana.js"></script>
</head>
<body>
<div id="top"><!-- do not remove this div, it is closed by doxygen! -->
<div id="titlearea">
<table cellspacing="0" cellpadding="0">
<tbody>
<tr style="height: 56px;">
 <td id="projectlogo"><img alt="Logo" src="Boost.png"/></td>
 <td style="padding-left: 0.5em;">
  <div id="projectname">Boost.Hana
   <span id="projectnumber">1.2.0</span>
  </div>
  <div id="projectbrief">Your standard library for metaprogramming</div>
 </td>
 <td>        <div id="MSearchBox" class="MSearchBoxInactive">
    <span class="left">
     <img id="MSearchSelect" src="search/mag_sel.png"
        onmouseover="return searchBox.OnSearchSelectShow()"
        onmouseout="return searchBox.OnSearchSelectHide()"
        alt=""/>
     <input type="text" id="MSearchField" value="Search" accesskey="S"
        onfocus="searchBox.OnSearchFieldFocus(true)"
        onblur="searchBox.OnSearchFieldFocus(false)"
        onkeyup="searchBox.OnSearchFieldChange(event)"/>
     </span><span class="right">
      <a id="MSearchClose" href="javascript:searchBox.CloseResultsWindow()"><img id="MSearchCloseImg"
border="0" src="search/close.png" alt=""/></a>
     </span>
```

```
        </div>
      </td>
    </tr>
  </tbody>
</table>
</div>
<!-- end header part -->
<!-- Generated by Doxygen 1.8.11 -->
<script type="text/javascript">
var searchBox = new SearchBox("searchBox", "search",false,'Search');
</script>
</div><!-- top -->
<div id="side-nav" class="ui-resizable side-nav-resizable">
 <div id="nav-tree">
   <div id="nav-tree-contents">
     <div id="nav-sync" class="sync"></div>
   </div>
 </div>
 <div id="splitbar" style="-moz-user-select:none;"
     class="ui-resizable-handle">
 </div>
</div>
<script type="text/javascript">
$(document).ready(function(){initNavTree('group__group-ext-mpl.html','');});
</script>
<div id="doc-content">
<!-- window showing the filter options -->
<div id="MSearchSelectWindow"
   onmouseover="return searchBox.OnSearchSelectShow()"
   onmouseout="return searchBox.OnSearchSelectHide()"
   onkeydown="return searchBox.OnSearchSelectKey(event)">
</div>

<!-- iframe showing the search results (closed by default) -->
<div id="MSearchResultsWindow">
<iframe src="javascript:void(0)" frameborder="0"
     name="MSearchResults" id="MSearchResults">
</iframe>
</div>

<div class="header">
 <div class="summary">
<a href="#nested-classes">Classes</a>  </div>
 <div class="headertitle">
<div class="title">Boost.MPL adapters<div class="ingroups"><a class="el" href="group__group-
ext.html">External adapters</a></div></div>  </div>
</div><!--header-->
<div class="contents">
```

```
<a name="details" id="details"></a><h2 class="groupheader">Description</h2>
<p>Adapters for Boost.MPL containers. </p>
<table class="memberdecls">
<tr class="heading"><td colspan="2"><h2 class="groupheader"><a name="nested-classes"></a>
Classes</h2></td></tr>
<tr class="memitem:"><td class="memItemLeft" align="right" valign="top">struct  </td><td
class="memItemRight" valign="bottom"><a class="el"
href="structboost_1_1mpl_1_1integral__c.html">boost::mpl::integral_c< T, v ></a></td></tr>
<tr class="memdesc:"><td class="mdescLeft"> </td><td class="mdescRight">Adapter for IntegralConstants
from the Boost.MPL.  <a href="structboost_1_1mpl_1_1integral__c.html#details">More...</a><br /></td></tr>
<tr class="separator:"><td class="memSeparator" colspan="2"> </td></tr>
<tr class="memitem:"><td class="memItemLeft" align="right" valign="top">struct  </td><td
class="memItemRight" valign="bottom"><a class="el" href="structboost_1_1mpl_1_1list.html">boost::mpl::list< T
></a></td></tr>
<tr class="memdesc:"><td class="mdescLeft"> </td><td class="mdescRight">Adapter for Boost.MPL lists.
<a href="structboost_1_1mpl_1_1list.html#details">More...</a><br /></td></tr>
<tr class="separator:"><td class="memSeparator" colspan="2"> </td></tr>
<tr class="memitem:"><td class="memItemLeft" align="right" valign="top">struct  </td><td
class="memItemRight" valign="bottom"><a class="el"
href="structboost_1_1mpl_1_1vector.html">boost::mpl::vector< T ></a></td></tr>
<tr class="memdesc:"><td class="mdescLeft"> </td><td class="mdescRight">Adapter for Boost.MPL
vectors.  <a href="structboost_1_1mpl_1_1vector.html#details">More...</a><br /></td></tr>
<tr class="separator:"><td class="memSeparator" colspan="2"> </td></tr>
</table>
</div><!-- contents -->
</div><!-- doc-content -->
<!--
Copyright Louis Dionne 2013-2017
Distributed under the Boost Software License, Version 1.0.
(See accompanying file LICENSE.md or copy at http://boost.org/LICENSE_1_0.txt)
-->
<!-- boost-no-inspect -->
<!-- HTML footer for doxygen 1.8.9.1-->
<!-- start footer part -->
<div id="nav-path" class="navpath"><!-- id is needed for treeview function! -->
 <ul>
 </ul>
</div>
</body>
</html>
var group__group_ext_mpl =
[
  [ "integral_c", "structboost_1_1mpl_1_1integral__c.html", null ],
  [ "list", "structboost_1_1mpl_1_1list.html", null ],
  [ "vector", "structboost_1_1mpl_1_1vector.html", null ]
];
#!/usr/bin/env python
from subprocess import check_output as run
```

```
from datetime import datetime
from itertools import groupby
from operator import itemgetter
import re
import magic


def authors(filename):
  log = run(['git', 'log', '--follow',
        '--date=short','--format=%aN%x09%ad', filename],
        universal_newlines=True)
  for line in log.splitlines():
    author, date = line.split('\t')
    if author != 'fix-copyright.py':
      yield author, datetime.strptime(date, '%Y-%m-%d')


def new_copyright(filename, previous):
  def f():
    au = list(authors(filename))
    alldates = map(itemgetter(1), au)
    aup = sorted(au + map(lambda a: (a, None), previous), key=itemgetter(0))
    for author, records in groupby(aup, itemgetter(0)):
      dates = filter(None, map(itemgetter(1), records))
      if not dates: dates = alldates
      start = min(dates)
      end = max(dates)
      fmt = '{0}' if start.year == end.year else '{0}-{1}'
      line = 'Copyright ' + fmt.format(start.year, end.year) + ' ' + author
      key = (start, author)
      yield key, line
  return map(itemgetter(1), sorted(f()))


def fix_copyright(filename):
  # Find copyright block in original file
  prefix = set()
  names = []
  lines = []
  with open(filename, 'r') as f:
    content = list(f)
  for i, line in enumerate(content[:15]):
    m = re.match(r'^(?P<prefix>\W*)(\(c\))?\s*?copyright\s*(\(c\))?\s+\d{4}(\s*-\s*\d{4})?\s+(?P<name>.+?)\s*$',
line, re.IGNORECASE)
    if m:
      d = m.groupdict()
      prefix.add(d['prefix'])
      lines.append(i)
      names.append(d['name'].strip())
  if len(prefix) != 1:
    print 'Not found:', filename
```

```
        return
    prefix = list(prefix)[0]

    print filename
    new = iter(new_copyright(filename, names))
    with open(filename, 'w') as f:
        for i, line in enumerate(content):
            if i in lines:
                for repl in new:
                    print >>f, prefix + repl
            else:
                print >>f, line,
    pass


def all_files():
    ls = run(['git', 'ls-files'], universal_newlines=True)
    for filename in ls.splitlines():
        if magic.from_file(filename, mime=True).split('/')[0] == 'text':
            yield filename


for f in all_files():
    fix_copyright(f)
```

[/===
  Boost.Geometry (aka GGL, Generic Geometry Library)

  Copyright (c) 2007-2013 Barend Gehrels, Amsterdam, the Netherlands.
  Copyright (c) 2008-2013 Bruno Lalande, Paris, France.
  Copyright (c) 2009-2013 Mateusz Loskot, London, UK.

  Use, modification and distribution is subject to the Boost Software License,
  Version 1.0. (See accompanying file LICENSE_1_0.txt or copy at
  http://www.boost.org/LICENSE_1_0.txt)
==/]
Use of this software is granted under one of the following two licenses,
to be chosen freely by the user.

1. Boost Software License - Version 1.0 - August 17th, 2003
==

Copyright (c) 2006, 2007 Marcin Kalicinski

Permission is hereby granted, free of charge, to any person or organization
obtaining a copy of the software and accompanying documentation covered by
this license (the "Software") to use, reproduce, display, distribute,
execute, and transmit the Software, and to prepare derivative works of the
Software, and to permit third-parties to whom the Software is furnished to
do so, all subject to the following:

The copyright notices in the Software and this entire statement, including
the above license grant, this restriction and the following disclaimer,
must be included in all copies of the Software, in whole or in part, and
all derivative works of the Software, unless such copies or derivative
works are solely in the form of machine-executable object code generated by
a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT
SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE
FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.

2. The MIT License
===

Copyright (c) 2006, 2007 Marcin Kalicinski

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL
THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
IN THE SOFTWARE.
===
Copyright (c) 2007-2011 Barend Gehrels, Amsterdam, the Netherlands.
Copyright (c) 2008-2011 Bruno Lalande, Paris, France.
Copyright (c) 2009-2011 Mateusz Loskot, London, UK.

Use, modification and distribution is subject to the Boost Software License,
Version 1.0. (See accompanying file LICENSE_1_0.txt or copy at
http://www.boost.org/LICENSE_1_0.txt)
===

The default copyright note for C++ source files reads:

// Boost.Geometry (aka GGL, Generic Geometry Library)

// Copyright (c) 2007-2011 Barend Gehrels, Amsterdam, the Netherlands.
// Copyright (c) 2008-2011 Bruno Lalande, Paris, France.
// Copyright (c) 2009-2011 Mateusz Loskot, London, UK.

// Parts of Boost.Geometry are redesigned from Geodan's Geographic Library
// (geolib/GGL), copyright (c) 1995-2010 Geodan, Amsterdam, the Netherlands.

// Use, modification and distribution is subject to the Boost Software License,
// Version 1.0. (See accompanying file LICENSE_1_0.txt or copy at
// http://www.boost.org/LICENSE_1_0.txt)

Exceptions:
1) Major work of any author -> might change order, change date
2) Exclusive work of one author including design -> might take sole copyright
 Examples:
  - adapted geometry types e.g. array, Boost.Polygon
  - implemented file-format e.g. WKB
  - implemented specific strategy or algorithm e.g. intersections
3) Examples -> might take sole copyright
4) Tests -> might take sole copyright
5) Docs -> might take sole copyright
6) Utilities e.g. converters -> might take sole copyright

The copyright note for other (source) files as .py, Jamfiles, etc is similar
and comments are changed accordingly.
<?xml version="1.0" encoding="utf-8"?>

<!-- Copyright (c) 2001-2006 CrystalClear Software, Inc.
    Subject to the Boost Software License, Version 1.0.
    (See accompanying file LICENSE_1_0.txt or http://www.boost.org/LICENSE_1_0.txt)
-->

<section id="date_time.license">
 <title>Boost Date-Time Library License</title>
 <bridgehead renderas="sect2">Boost Date-Time Library License</bridgehead>
 <!--
 <ulink url="../../../index.htm">
   <imagedata align="left"
         format="GIF"
      fileref="../../../boost.png"
      alt="C++ Boost" />
   </ulink>-->
 <!--hr /-->

```
<para>
  The following is the overall license for the boost date_time
  library. This notice is found in all source files related to the
  library.
</para>
<para>
  Copyright &#169; 2002 CrystalClear Software, Inc.
</para>
<para>
  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. CrystalClear Software makes no
  representations about the suitability of this software for any
  purpose. It is provided "as is" without express or implied
  warranty.
</para>
<!--hr / -->
<ulink url="mailto:jeff@crystalclearsoftware.com">Jeff Garland</ulink>
&#169; 2000-2002
</section>
[/
 Copyright 2007 John Maddock.
 Distributed under the Boost Software License, Version 1.0.
 (See accompanying file LICENSE_1_0.txt or copy at
 http://www.boost.org/LICENSE_1_0.txt).
]
```

[section:mpl MPL Interoperability]

All the value based traits in this library conform to MPL's requirements
for an [@../../../../libs/mpl/doc/refmanual/integral-constant.html Integral Constant type].

Please note that these types no longer inherit from `mpl::true_` or `mpl::false_` etc, and the library
will no longer implicitly include any MPL header. However there is an implicit conversion from
`integral_constant` to the corresponding MPL types, therefore tag-dispatching that uses MPL types
in function overloads will still work as before.

[endsect]
```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=US-ASCII">
<title>MPL Interoperability</title>
<link rel="stylesheet" href="../../../../../doc/src/boostbook.css" type="text/css">
<meta name="generator" content="DocBook XSL Stylesheets V1.79.1">
<link rel="home" href="../index.html" title="Chapter 1. Boost.TypeTraits">
<link rel="up" href="../index.html" title="Chapter 1. Boost.TypeTraits">
```

```
<link rel="prev" href="intrinsics.html" title="Support for Compiler Intrinsics">
<link rel="next" href="examples.html" title="Examples">
</head>
<body bgcolor="white" text="black" link="#0000FF" vlink="#840084" alink="#0000FF">
<table cellpadding="2" width="100%"><tr>
<td valign="top"><img alt="Boost C++ Libraries" width="277" height="86" src="../../../../../boost.png"></td>
<td align="center"><a href="../../../../../index.html">Home</a></td>
<td align="center"><a href="../../../../../libs/libraries.htm">Libraries</a></td>
<td align="center"><a href="http://www.boost.org/users/people.html">People</a></td>
<td align="center"><a href="http://www.boost.org/users/faq.html">FAQ</a></td>
<td align="center"><a href="../../../../../more/index.htm">More</a></td>
</tr></table>
<hr>
<div class="spirit-nav">
<a accesskey="p" href="intrinsics.html"><img src="../../../../../doc/src/images/prev.png" alt="Prev"></a><a
accesskey="u" href="../index.html"><img src="../../../../../doc/src/images/up.png" alt="Up"></a><a accesskey="h"
href="../index.html"><img src="../../../../../doc/src/images/home.png" alt="Home"></a><a accesskey="n"
href="examples.html"><img src="../../../../../doc/src/images/next.png" alt="Next"></a>
</div>
<div class="section">
<div class="titlepage"><div><div><h2 class="title" style="clear: both">
<a name="boost_typetraits.mpl"></a><a class="link" href="mpl.html" title="MPL Interoperability">MPL
Interoperability</a>
</h2></div></div></div>
<p>
    All the value based traits in this library conform to MPL's requirements for
    an <a href="../../../../../libs/mpl/doc/refmanual/integral-constant.html" target="_top">Integral
    Constant type</a>.
  </p>
<p>
    Please note that these types no longer inherit from <code class="computeroutput"><span
class="identifier">mpl</span><span class="special">::</span><span class="identifier">true_</span></code> or
    <code class="computeroutput"><span class="identifier">mpl</span><span class="special">::</span><span
class="identifier">false_</span></code> etc, and the library will no longer
    implicitly include any MPL header. However there is an implicit conversion
    from <code class="computeroutput"><span class="identifier">integral_constant</span></code> to the
    corresponding MPL types, therefore tag-dispatching that uses MPL types in function
    overloads will still work as before.
  </p>
</div>
<table xmlns:rev="http://www.cs.rpi.edu/~gregod/boost/tools/doc/revision" width="100%"><tr>
<td align="left"></td>
<td align="right"><div class="copyright-footer">Copyright &#169; 2000, 2011 Adobe Systems Inc, David
Abrahams,
    Frederic Bron, Steve Cleary, Beman Dawes, Aleksey Gurtovoy, Howard Hinnant,
    Jesse Jones, Mat Marcus, Itay Maman, John Maddock, Alexander Nasonov, Thorsten
    Ottosen, Roman Perepelitsa, Robert Ramey, Jeremy Siek, Robert Stewart and Steven
    Watanabe<p>
```

```
        Distributed under the Boost Software License, Version 1.0. (See accompanying
        file LICENSE_1_0.txt or copy at <a href="http://www.boost.org/LICENSE_1_0.txt"
target="_top">http://www.boost.org/LICENSE_1_0.txt</a>)
        </p>
</div></td>
</tr></table>
<hr>
<div class="spirit-nav">
<a accesskey="p" href="intrinsics.html"><img src="../../../../../doc/src/images/prev.png" alt="Prev"></a><a
accesskey="u" href="../index.html"><img src="../../../../../doc/src/images/up.png" alt="Up"></a><a accesskey="h"
href="../index.html"><img src="../../../../../doc/src/images/home.png" alt="Home"></a><a accesskey="n"
href="examples.html"><img src="../../../../../doc/src/images/next.png" alt="Next"></a>
</div>
</body>
</html>
[/
 Copyright 2006 - 2010 John Maddock and Paul A. Bristow.
 Distributed under the Boost Software License, Version 1.0.
 (See accompanying file LICENSE_1_0.txt or copy at
 http://www.boost.org/LICENSE_1_0.txt).
]
//////////////////////////////////////////////////////////////////////////////
//
// (C) Copyright Ion Gaztanaga 2005-2013.
//
// Distributed under the Boost Software License, Version 1.0.
// (See accompanying file LICENSE_1_0.txt or copy at
// http://www.boost.org/LICENSE_1_0.txt)
//
// See http://www.boost.org/libs/container for documentation.
//
//////////////////////////////////////////////////////////////////////////////

#ifndef BOOST_CONTAINER_CONTAINER_DETAIL_MPL_HPP
#define BOOST_CONTAINER_CONTAINER_DETAIL_MPL_HPP

#ifndef BOOST_CONFIG_HPP
#  include <boost/config.hpp>
#endif

#if defined(BOOST_HAS_PRAGMA_ONCE)
#  pragma once
#endif

#include <boost/container/detail/config_begin.hpp>
#include <boost/container/detail/workaround.hpp>
#include <boost/move/detail/type_traits.hpp>
#include <boost/intrusive/detail/mpl.hpp>
```

```
#include <cstddef>

namespace boost {
namespace container {
namespace container_detail {

using boost::move_detail::integral_constant;
using boost::move_detail::true_type;
using boost::move_detail::false_type;
using boost::move_detail::enable_if_c;
using boost::move_detail::enable_if;
using boost::move_detail::enable_if_convertible;
using boost::move_detail::disable_if_c;
using boost::move_detail::disable_if;
using boost::move_detail::disable_if_convertible;
using boost::move_detail::is_convertible;
using boost::move_detail::if_c;
using boost::move_detail::if_;
using boost::move_detail::identity;
using boost::move_detail::bool_;
using boost::move_detail::true_;
using boost::move_detail::false_;
using boost::move_detail::yes_type;
using boost::move_detail::no_type;
using boost::move_detail::bool_;
using boost::move_detail::true_;
using boost::move_detail::false_;
using boost::move_detail::unvoid_ref;
using boost::move_detail::and_;
using boost::move_detail::or_;
using boost::move_detail::not_;
using boost::move_detail::enable_if_and;
using boost::move_detail::disable_if_and;
using boost::move_detail::enable_if_or;
using boost::move_detail::disable_if_or;

template <class FirstType>
struct select1st
{
  typedef FirstType type;

  template<class T>
  const type& operator()(const T& x) const
  {  return x.first;   }

  template<class T>
  type& operator()(T& x)
```

```
   {  return const_cast<type&>(x.first);   }
};


} //namespace container_detail {
} //namespace container {
} //namespace boost {

#include <boost/container/detail/config_end.hpp>

#endif   //#ifndef BOOST_CONTAINER_CONTAINER_DETAIL_MPL_HPP
//////////////////////////////////////////////////////////////////////////////
//
// (C) Copyright Ion Gaztanaga 2005-2016.
//
// Distributed under the Boost Software License, Version 1.0.
// (See accompanying file LICENSE_1_0.txt or copy at
// http://www.boost.org/LICENSE_1_0.txt)
//
// See http://www.boost.org/libs/interprocess for documentation.
//
//////////////////////////////////////////////////////////////////////////////

#ifndef BOOST_INTERPROCESS_DETAIL_MPL_HPP
#define BOOST_INTERPROCESS_DETAIL_MPL_HPP

#ifndef BOOST_CONFIG_HPP
#  include <boost/config.hpp>
#endif
#
#if defined(BOOST_HAS_PRAGMA_ONCE)
#  pragma once
#endif

#include <cstddef>

namespace boost {
namespace interprocess {
namespace ipcdetail {

template <class T, T val>
struct integral_constant
{
  static const T value = val;
  typedef integral_constant<T,val> type;
};

template< bool C_ >
struct bool_ : integral_constant<bool, C_>
```

```cpp
{
 static const bool value = C_;
};

typedef bool_<true>        true_;
typedef bool_<false>       false_;

typedef true_  true_type;
typedef false_ false_type;

typedef char yes_type;
struct no_type
{
 char padding[8];
};

template <bool B, class T = void>
struct enable_if_c {
 typedef T type;
};

template <class T>
struct enable_if_c<false, T> {};

template <class Cond, class T = void>
struct enable_if : public enable_if_c<Cond::value, T> {};

template <class Cond, class T = void>
struct disable_if : public enable_if_c<!Cond::value, T> {};

template<
    bool C
  , typename T1
  , typename T2
  >
struct if_c
{
  typedef T1 type;
};

template<
    typename T1
  , typename T2
  >
struct if_c<false,T1,T2>
{
  typedef T2 type;
};
```

```
template<
    typename T1
  , typename T2
  , typename T3
  >
struct if_
{
 typedef typename if_c<0 != T1::value, T2, T3>::type type;
};


template<std::size_t S>
struct ls_zeros
{
 static const std::size_t value = (S & std::size_t(1)) ? 0 : (1u + ls_zeros<(S >> 1u)>::value);
};

template<>
struct ls_zeros<0>
{
 static const std::size_t value = 0;
};

template<>
struct ls_zeros<1>
{
 static const std::size_t value = 0;
};

} //namespace ipcdetail {
} //namespace interprocess {
} //namespace boost {

#endif   //#ifndef BOOST_INTERPROCESS_DETAIL_MPL_HPP
//////////////////////////////////////////////////////////////////////////////
//
// (C) Copyright Ion Gaztanaga      2006-2014
// (C) Copyright Microsoft Corporation  2014
//
// Distributed under the Boost Software License, Version 1.0.
//    (See accompanying file LICENSE_1_0.txt or copy at
//         http://www.boost.org/LICENSE_1_0.txt)
//
// See http://www.boost.org/libs/intrusive for documentation.
//
//////////////////////////////////////////////////////////////////////////////
```

```
#ifndef BOOST_INTRUSIVE_DETAIL_MPL_HPP
#define BOOST_INTRUSIVE_DETAIL_MPL_HPP

#ifndef BOOST_CONFIG_HPP
#  include <boost/config.hpp>
#endif

#if defined(BOOST_HAS_PRAGMA_ONCE)
#  pragma once
#endif

#include <boost/intrusive/detail/config_begin.hpp>
#include <boost/move/detail/type_traits.hpp>
#include <cstddef>

namespace boost {
namespace intrusive {
namespace detail {

using boost::move_detail::is_same;
using boost::move_detail::add_const;
using boost::move_detail::remove_const;
using boost::move_detail::remove_cv;
using boost::move_detail::remove_reference;
using boost::move_detail::add_reference;
using boost::move_detail::remove_pointer;
using boost::move_detail::add_pointer;
using boost::move_detail::true_type;
using boost::move_detail::false_type;
using boost::move_detail::enable_if_c;
using boost::move_detail::enable_if;
using boost::move_detail::disable_if_c;
using boost::move_detail::disable_if;
using boost::move_detail::is_convertible;
using boost::move_detail::if_c;
using boost::move_detail::if_;
using boost::move_detail::is_const;
using boost::move_detail::identity;
using boost::move_detail::alignment_of;
using boost::move_detail::is_empty;
using boost::move_detail::addressof;
using boost::move_detail::integral_constant;
using boost::move_detail::enable_if_convertible;
using boost::move_detail::disable_if_convertible;
using boost::move_detail::bool_;
using boost::move_detail::true_;
using boost::move_detail::false_;
using boost::move_detail::yes_type;
```

```cpp
using boost::move_detail::no_type;
using boost::move_detail::apply;
using boost::move_detail::eval_if_c;
using boost::move_detail::eval_if;
using boost::move_detail::unvoid_ref;
using boost::move_detail::add_const_if_c;

template<std::size_t S>
struct ls_zeros
{
  static const std::size_t value = (S & std::size_t(1)) ? 0 : (1 + ls_zeros<(S>>1u)>::value);
};

template<>
struct ls_zeros<0>
{
  static const std::size_t value = 0;
};

template<>
struct ls_zeros<1>
{
  static const std::size_t value = 0;
};

// Infrastructure for providing a default type for T::TNAME if absent.
#define BOOST_INTRUSIVE_INSTANTIATE_DEFAULT_TYPE_TMPLT(TNAME)     \
  template <typename T, typename DefaultType>                  \
  struct boost_intrusive_default_type_ ## TNAME               \
  {                                                 \
    template <typename X>                           \
    static char test(int, typename X::TNAME*);               \
                                                    \
    template <typename X>                           \
    static int test(...);                           \
                                                    \
    struct DefaultWrap { typedef DefaultType TNAME; };        \
                                                    \
    static const bool value = (1 == sizeof(test<T>(0, 0)));    \
                                                    \
    typedef typename                               \
      ::boost::intrusive::detail::if_c               \
        <value, T, DefaultWrap>::type::TNAME type;        \
  };                                                \
  //

#define BOOST_INTRUSIVE_OBTAIN_TYPE_WITH_DEFAULT(INSTANTIATION_NS_PREFIX, T,
TNAME, TIMPL)  \
```

```
    typename INSTANTIATION_NS_PREFIX                                          \
      boost_intrusive_default_type_ ## TNAME< T, TIMPL >::type                \
//

#define BOOST_INTRUSIVE_INSTANTIATE_EVAL_DEFAULT_TYPE_TMPLT(TNAME)\
  template <typename T, typename DefaultType>            \
  struct boost_intrusive_eval_default_type_ ## TNAME          \
  {                                                                 \
    template <typename X>                          \
    static char test(int, typename X::TNAME*);            \
                                                            \
    template <typename X>                          \
    static int test(...);                          \
                                                            \
    struct DefaultWrap                       \
    { typedef typename DefaultType::type TNAME; };         \
                                                            \
    static const bool value = (1 == sizeof(test<T>(0, 0)));    \
                                                            \
    typedef typename                          \
      ::boost::intrusive::detail::eval_if_c              \
        < value                               \
        , ::boost::intrusive::detail::identity<T>         \
        , ::boost::intrusive::detail::identity<DefaultWrap>  \
        >::type::TNAME type;                       \
  };                                                    \
//

#define BOOST_INTRUSIVE_OBTAIN_TYPE_WITH_EVAL_DEFAULT(INSTANTIATION_NS_PREFIX, T,
TNAME, TIMPL) \
    typename INSTANTIATION_NS_PREFIX                                          \
      boost_intrusive_eval_default_type_ ## TNAME< T, TIMPL >::type           \
//

#define BOOST_INTRUSIVE_INTERNAL_STATIC_BOOL_IS_TRUE(TRAITS_PREFIX,
TYPEDEF_TO_FIND) \
template <class T>\
struct TRAITS_PREFIX##_bool\
{\
  template<bool Add>\
  struct two_or_three {yes_type _[2 + Add];};\
  template <class U> static yes_type test(...);\
  template <class U> static two_or_three<U::TYPEDEF_TO_FIND> test (int);\
  static const std::size_t value = sizeof(test<T>(0));\
};\
\
template <class T>\
struct TRAITS_PREFIX##_bool_is_true\
```

```
{\
  static const bool value = TRAITS_PREFIX##_bool<T>::value > sizeof(yes_type)*2;\
};\
//

#define BOOST_INTRUSIVE_HAS_STATIC_MEMBER_FUNC_SIGNATURE(TRAITS_NAME, \
FUNC_NAME) \
  template <typename U, typename Signature> \
  class TRAITS_NAME \
  { \
  private: \
  template<Signature> struct helper;\
  template<typename T> \
  static ::boost::intrusive::detail::yes_type test(helper<&T::FUNC_NAME>*); \
  template<typename T> static ::boost::intrusive::detail::no_type test(...); \
  public: \
  static const bool value = sizeof(test<U>(0)) == sizeof(::boost::intrusive::detail::yes_type); \
  }; \
//

#define BOOST_INTRUSIVE_HAS_MEMBER_FUNC_CALLED(TRAITS_NAME, FUNC_NAME) \
template <typename Type> \
struct TRAITS_NAME \
{ \
  struct BaseMixin \
  { \
    void FUNC_NAME(); \
  }; \
  struct Base : public Type, public BaseMixin { Base(); }; \
  template <typename T, T t> class Helper{}; \
  template <typename U> \
  static ::boost::intrusive::detail::no_type  test(U*, Helper<void (BaseMixin::*)(), &U::FUNC_NAME>* = 0); \
  static ::boost::intrusive::detail::yes_type test(...); \
  static const bool value = sizeof(::boost::intrusive::detail::yes_type) == sizeof(test((Base*)(0))); \
};\
//

#define BOOST_INTRUSIVE_HAS_MEMBER_FUNC_CALLED_IGNORE_SIGNATURE(TRAITS_NAME, \
FUNC_NAME) \
BOOST_INTRUSIVE_HAS_MEMBER_FUNC_CALLED(TRAITS_NAME##_ignore_signature, FUNC_NAME) \
\
\
template <typename Type, class> \
struct TRAITS_NAME \
  : public TRAITS_NAME##_ignore_signature<Type> \
{};\
//
```

```
} //namespace detail
} //namespace intrusive
} //namespace boost

#include <boost/intrusive/detail/config_end.hpp>

#endif //BOOST_INTRUSIVE_DETAIL_MPL_HPP
```

1.2 breakpad r1214

1.2.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or

Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor 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, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work

or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
Work or Derivative Works thereof in any medium, with or without
modifications, and in Source or Object form, provided that You
meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work

by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [2007] Neal Norwitz
Portions Copyright [2007] Google Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

  * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
  * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
  * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY

THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 2006, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

    * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
    * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
    * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1996 - 2011, Daniel Stenberg, <daniel@haxx.se>.

All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose
with or without fee is hereby granted, provided that the above copyright
notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN
NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE
OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

This package was debianized by Google Inc. <opensource@google.com> on 13 June 2008.

It was downloaded from http://code.google.com/

Upstream Author: opensource@google.com

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
  * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
  * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  * Redistributions of source code must retain the above copyright

notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
   * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

A function gettimeofday in utilities.cc is based on

http://www.google.com/codesearch/p?hl=en#dR3YEbitojA/COPYING&q=GetSystemTimeAsFileTime%20license:
bsd

The license of this code is:

Copyright (c) 2003-2008, Jouni Malinen <j@w1.fi> and contributors
All Rights Reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.

3. Neither the name(s) of the above-listed copyright holder(s) nor the
   names of its contributors may be used to endorse or promote products
   derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS

"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

    * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
    * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
    * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

1.3 Expat 2.2.0 :2.2.0

1.3.1 Available under license :

Copyright (c) 1998-2000 Thai Open Source Software Center Ltd and Clark Cooper

Copyright (c) 2001-2016 Expat maintainers

Permission is hereby granted, free of charge, to any person obtaining

a copy of this software and associated documentation files (the

"Software"), to deal in the Software without restriction, including

without limitation the rights to use, copy, modify, merge, publish,

distribute, sublicense, and/or sell copies of the Software, and to

permit persons to whom the Software is furnished to do so, subject to

the following conditions:

The above copyright notice and this permission notice shall be included

in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,

EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF

MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.

IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY

CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,

TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE

SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.4 glib 2.27.1

1.4.1 Available under license :

GNU LIBRARY GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.

59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies

of this license document, but changing it is not allowed.

[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

    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., 59 Temple Place - Suite 330,
Boston, MA 02111-1307 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!
This package was debianized by Ben Gertzfield <che@debian.org>
on Tue, 17 Nov 1998 00:16:10 -0800

It was downloaded from the CVS archives at cvs.gimp.org.

It may be redistributed under the terms of the GNU LGPL, Version 2 or
later, found on Debian systems in the file /usr/doc/copyright/LGPL.
This work may be reproduced and distributed in whole or in part, in
any medium, physical or electronic, so as long as this copyright
notice remains intact and unchanged on all copies. Commercial
redistribution is permitted and encouraged, but you may not
redistribute, in whole or in part, under terms more restrictive than
those under which you received it. If you redistribute a modified or
translated version of this work, you must also make the source code to
the modified or translated version available in electronic form
without charge. However, mere aggregation as part of a larger work

shall not count as a modification for this purpose.

All code examples in this work are placed into the public domain,
and may be used, modified and redistributed without restriction.

BECAUSE THIS WORK IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE WORK, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE WORK "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. SHOULD THE WORK PROVE DEFECTIVE, YOU ASSUME
THE COST OF ALL NECESSARY REPAIR OR CORRECTION.

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 WORK 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
WORK, EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.
PCRE LICENCE

Please see the file LICENCE in the PCRE distribution for licensing details.

End

1.5 gstreamer 0.10.35.1

1.5.1 Available under license :

GNU LIBRARY GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
675 Mass Ave, Cambridge, MA 02139, USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[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!
<sect1 id="chapter-legal">
 <title id="title-legal">GStreamer Legal Issues</title>
 <para>
This part of the FAQ is based on a series of questions we asked the FSF
to understand how the GPL works and how patents affects the GPL. These
questions were answered by the <ulink url="http://www.fsf.org/">
FSF lawyers</ulink>, so we view them as the
final interpretation on how the GPL and LGPL interact with patents in our
opinion. This consultancy was paid for by
<ulink url="http://www.fluendo.com/">Fluendo</ulink>
in order to obtain clear and quotable answers. These answers were certified
by the FSF lawyer team and verified by FSF lawyer and law professor Eben Moglen.
 </para>

 <qandaset defaultlabel="qanda">

```
  <qandaentry>
    <question id="legal-distribute-three">
      <para>
```
Can someone distribute the combination of
```
<itemizedlist>
 <listitem><para>GStreamer, the LGPL library</para></listitem>
 <listitem><para>MyPlayer, a GPL playback application</para></listitem>
 <listitem><para>The binary-only Sorenson decoder</para></listitem>
</itemizedlist>
```
together in one distribution/operating system ? If not, what
needs to be changed to make this possible ?
```
      </para>
    </question>

    <answer>
      <para>
```
This would be a problem, because the GStreamer and MyPlayer licenses would
forbid it. In order to link GStreamer to MyPlayer, you need to use section
3 of the LGPL to convert GStreamer to GPL. The GPL version of GStreamer
forbids linking to the Sorenson decoder. Anyway, the MyPlayer GPL
license forbids this.
```
      </para>

      <para>
```
If the authors of MyPlayer want to permit this, we have an
exception for them: the controlled interface exception from the FAQ.
The idea of this is that you can't get around the GPL just by including
a LGPL bit in the middle.
```
      </para>

      <para>
```
Note: MyPlayer is a completely fictituous application at the time of writing.
```
      </para>

    </answer>
  </qandaentry>

  <qandaentry>
    <question id="legal-binary-plugin">
      <para>
```
Suppose Apple wants to write a binary-only proprietary
plugin for GStreamer to decode Sorenson video, which will be shipped
stand-alone, not part of a package like in the question above.
Can Apple distribute this binary-only plugin ?
```
      </para>
    </question>

    <answer>
```

```
        <para>
Yes, modulo certain reverse engineering requirements in section 6 of
the LGPL.
        </para>
      </answer>
    </qandaentry>


    <qandaentry>
      <question id="legal-gpl-program">
        <para>
If a program released under the GPL uses a library that
is LGPL, and this library can dlopen plug-ins at runtime, what are the
requirements for the license of the plug-in ?
        </para>
      </question>


      <answer>
        <para>
You may not distribute the plug-in with the GPL application.
Distributing the plug-in alone, with the knowledge that it will be used
primarily by GPL software is a bit of an edge case.  We will not advise you
that it would be safe to do so, but we also will not advise you that it
would be absolutely forbidden.
        </para>
      </answer>
    </qandaentry>


    <qandaentry>
      <question id="legal-safe-countries">
        <para>
Can someone in a country that does not have software patents distribute
code covered by US patents under the GPL to people in, for example, Norway ?
If he/she visits the US, can he/she be arrested ?
        </para>
      </question>


      <answer>
        <para>
Yes, he can.
No,  there are no criminal penalties for patent infringement in the US.
        </para>
      </answer>
    </qandaentry>


    <qandaentry>
      <question id="legal-unsafe-countries">
        <para>
Can someone from the US distribute software covered by
```

US patents under the GPL to people in Norway ? To people in the US ?
    </para>
    </question>

    <answer>
    <para>
This might infringe some patents, but the GPL would not forbid it
absent some actual restriction, such as a court judgement or agreement.
The US government is empowered to refuse importation of patent
infringing devices, including software.
    </para>
    </answer>
  </qandaentry>

  <qandaentry>
    <question id="legal-gpl-library-patents">
    <para>
There are a lot of GPL- or LGPL-licensed libraries that
handle media codecs which have patents. Take mad, an mp3 decoding library,
as an example. It is licensed under the GPL. In countries where patents
are valid, does this invalidate the GPL license for this project ?
    </para>
    </question>

    <answer>
    <para>
The mere existence of a patent which might read on the program does not
change anything. However, if a court judgement or other agreement
prevents you from distributing libmad under GPL terms, you can not
distribute it at all.
    </para>

    <para>
The GPL and LGPL say (sections 7 and 11):
<quote>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.</quote>
    </para>
    </answer>
  </qandaentry>

  <qandaentry>
    <question id="legal-gpl-court-judgment">
    <para>
So let's say there is a court judgement. Does this mean that the GPL license is
invalid for the project everywhere, or only in the countries where it conflicts
with the applicable patents ?
    </para>

```
      </question>

      <answer>
        <para>
The GPL operates on a per-action, not per-program basis.  That is, if
you are in a country which has software patents, and a court tells you
that you cannot distribute (say) libmad in source code form, then you
cannot distribute libmad at all.  This doesn't affect anyone else.
        </para>
      </answer>
    </qandaentry>

    <qandaentry>
      <question id="legal-gpl-and-binary">
        <para>
Patented decoding can be implemented in GStreamer either by
having a binary-only plugin do the decoding, or by writing a plugin
(with any applicable license) that links to a binary-only library.
Does this affect the licensing issues involved in regards to GPL/LGPL?
        </para>
      </question>

      <answer>
        <para>
No.
        </para>
      </answer>
    </qandaentry>

    <qandaentry>
      <question id="legal-gpl-patent-distribution">
        <para>
Is it correct that you cannot distribute the GPL mad library to
decode mp3's, *even* in the case where you have obtained a valid license
for decoding mp3 ?
        </para>
      </question>

      <answer>
        <para>
The only GPL-compatible patent licenses are those which are open to
all parties posessing copies of GPL software which practices the
teachings of the patent.
        </para>

        <para>
If you take a license which doesn't allow others to distribute
original or modified versions of libmad practicing the same patent
```

claims as the version you distribute, then you may not distribute at
all.
　　　</para>
　　</answer>
　</qandaentry>
　</qandaset>
</sect1>
/* GStreamer
* Copyright (C) <1999> Erik Walthinsen <omega@cse.ogi.edu>
*
* 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., 59 Temple Place - Suite 330,
* Boston, MA 02111-1307, USA.
*/
/* GStreamer
* Copyright (C) 1999,2000 Erik Walthinsen <omega@cse.ogi.edu>
*   2000 Wim Taymans <wtay@chello.be>
*
* gstpadtemplate.c: Templates for pad creation
*
* 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., 59 Temple Place - Suite 330,
* Boston, MA 02111-1307, USA.
*/

/**

```
* SECTION:gstpadtemplate
* @short_description: Describe the media type of a pad.
* @see_also: #GstPad, #GstElementFactory
*
* Padtemplates describe the possible media types a pad or an elementfactory can
* handle. This allows for both inspection of handled types before loading the
* element plugin as well as identifying pads on elements that are not yet
* created (request or sometimes pads).
*
* Pad and PadTemplates have #GstCaps attached to it to describe the media type
* they are capable of dealing with. gst_pad_template_get_caps() or
* GST_PAD_TEMPLATE_CAPS() are used to get the caps of a padtemplate. It's not
* possible to modify the caps of a padtemplate after creation.
*
* PadTemplates have a #GstPadPresence property which identifies the lifetime
* of the pad and that can be retrieved with GST_PAD_TEMPLATE_PRESENCE(). Also
* the direction of the pad can be retrieved from the #GstPadTemplate with
* GST_PAD_TEMPLATE_DIRECTION().
*
* The GST_PAD_TEMPLATE_NAME_TEMPLATE () is important for GST_PAD_REQUEST pads
* because it has to be used as the name in the gst_element_get_request_pad()
* call to instantiate a pad from this template.
*
* Padtemplates can be created with gst_pad_template_new() or with
* gst_static_pad_template_get (), which creates a #GstPadTemplate from a
* #GstStaticPadTemplate that can be filled with the
* convenient GST_STATIC_PAD_TEMPLATE() macro.
*
* A padtemplate can be used to create a pad (see gst_pad_new_from_template()
* or gst_pad_new_from_static_template ()) or to add to an element class
* (see gst_element_class_add_pad_template ()).
*
* The following code example shows the code to create a pad from a padtemplate.
* <example>
* <title>Create a pad from a padtemplate</title>
*   <programlisting>
*   GstStaticPadTemplate my_template =
*   GST_STATIC_PAD_TEMPLATE (
*     "sink",          // the name of the pad
*     GST_PAD_SINK,    // the direction of the pad
*     GST_PAD_ALWAYS,  // when this pad will be present
*     GST_STATIC_CAPS (       // the capabilities of the padtemplate
*       "audio/x-raw-int, "
*        "channels = (int) [ 1, 6 ]"
*     )
*   );
*   void
*   my_method (void)
```

```
*   {
*     GstPad *pad;
*     pad = gst_pad_new_from_static_template (&amp;my_template, "sink");
*     ...
*   }
*   </programlisting>
* </example>
*
* The following example shows you how to add the padtemplate to an
* element class, this is usually done in the base_init of the class:
* <informalexample>
*   <programlisting>
*   static void
*   my_element_base_init (gpointer g_class)
*   {
*     GstElementClass *gstelement_class = GST_ELEMENT_CLASS (g_class);
*
*     gst_element_class_add_pad_template (gstelement_class,
*         gst_static_pad_template_get (&amp;my_template));
*   }
*   </programlisting>
* </informalexample>
*
* Last reviewed on 2006-02-14 (0.10.3)
*/

#include "gst_private.h"

#include "gstpad.h"
#include "gstpadtemplate.h"
#include "gstenumtypes.h"
#include "gstmarshal.h"
#include "gstutils.h"
#include "gstinfo.h"
#include "gsterror.h"
#include "gstvalue.h"

#define GST_CAT_DEFAULT GST_CAT_PADS

enum
{
  PROP_NAME_TEMPLATE = 1,
  PROP_DIRECTION,
  PROP_PRESENCE,
  PROP_CAPS
};

enum
```

```
{
TEMPL_PAD_CREATED,
/* FILL ME */
LAST_SIGNAL
};

static GstObject *parent_class = NULL;
static guint gst_pad_template_signals[LAST_SIGNAL] = { 0 };

static void gst_pad_template_dispose (GObject * object);
static void gst_pad_template_set_property (GObject * object, guint prop_id,
  const GValue * value, GParamSpec * pspec);
static void gst_pad_template_get_property (GObject * object, guint prop_id,
  GValue * value, GParamSpec * pspec);

G_DEFINE_TYPE (GstPadTemplate, gst_pad_template, GST_TYPE_OBJECT);

static void
gst_pad_template_class_init (GstPadTemplateClass * klass)
{
GObjectClass *gobject_class;
GstObjectClass *gstobject_class;

gobject_class = (GObjectClass *) klass;
gstobject_class = (GstObjectClass *) klass;

parent_class = g_type_class_peek_parent (klass);

/**
 * GstPadTemplate::pad-created:
 * @pad_template: the object which received the signal.
 * @pad: the pad that was created.
 *
 * This signal is fired when an element creates a pad from this template.
 */
gst_pad_template_signals[TEMPL_PAD_CREATED] =
    g_signal_new ("pad-created", G_TYPE_FROM_CLASS (klass), G_SIGNAL_RUN_LAST,
    G_STRUCT_OFFSET (GstPadTemplateClass, pad_created),
    NULL, NULL, gst_marshal_VOID__OBJECT, G_TYPE_NONE, 1, GST_TYPE_PAD);

gobject_class->dispose = gst_pad_template_dispose;

gobject_class->get_property = gst_pad_template_get_property;
gobject_class->set_property = gst_pad_template_set_property;

/**
 * GstPadTemplate:name-template
 *
```

```
 * The name template of the pad template.
 *
 * Since: 0.10.21
 */
g_object_class_install_property (gobject_class, PROP_NAME_TEMPLATE,
    g_param_spec_string ("name-template", "Name template",
        "The name template of the pad template", NULL,
        G_PARAM_READWRITE | G_PARAM_CONSTRUCT_ONLY | G_PARAM_STATIC_STRINGS));


/**
 * GstPadTemplate:direction
 *
 * The direction of the pad described by the pad template.
 *
 * Since: 0.10.21
 */
g_object_class_install_property (gobject_class, PROP_DIRECTION,
    g_param_spec_enum ("direction", "Direction",
        "The direction of the pad described by the pad template",
        GST_TYPE_PAD_DIRECTION, GST_PAD_UNKNOWN,
        G_PARAM_READWRITE | G_PARAM_CONSTRUCT_ONLY | G_PARAM_STATIC_STRINGS));


/**
 * GstPadTemplate:presence
 *
 * When the pad described by the pad template will become available.
 *
 * Since: 0.10.21
 */
g_object_class_install_property (gobject_class, PROP_PRESENCE,
    g_param_spec_enum ("presence", "Presence",
        "When the pad described by the pad template will become available",
        GST_TYPE_PAD_PRESENCE, GST_PAD_ALWAYS,
        G_PARAM_READWRITE | G_PARAM_CONSTRUCT_ONLY | G_PARAM_STATIC_STRINGS));


/**
 * GstPadTemplate:caps
 *
 * The capabilities of the pad described by the pad template.
 *
 * Since: 0.10.21
 */
g_object_class_install_property (gobject_class, PROP_CAPS,
    g_param_spec_boxed ("caps", "Caps",
        "The capabilities of the pad described by the pad template",
        GST_TYPE_CAPS,
        G_PARAM_READWRITE | G_PARAM_CONSTRUCT_ONLY | G_PARAM_STATIC_STRINGS));
```

```
  gstobject_class->path_string_separator = "*";
}

static void
gst_pad_template_init (GstPadTemplate * templ)
{
 /* FIXME 0.11: Does anybody remember why this is here? If not, let's
  * change it for 0.11 and let gst_element_class_add_pad_template() for
  * example ref/sink the pad templates.
  */
 /* We ensure that the pad template we're creating has a sunken reference.
  * Inconsistencies in pad templates being floating or sunken has caused
  * problems in the past with leaks, etc.
  *
  * For consistency, then, we only produce them  with sunken references
  * owned by the creator of the object
  */
 if (GST_OBJECT_IS_FLOATING (templ)) {
  gst_object_ref_sink (templ);
 }
}

static void
gst_pad_template_dispose (GObject * object)
{
 GstPadTemplate *templ = GST_PAD_TEMPLATE (object);

 g_free (GST_PAD_TEMPLATE_NAME_TEMPLATE (templ));
 if (GST_PAD_TEMPLATE_CAPS (templ)) {
  gst_caps_unref (GST_PAD_TEMPLATE_CAPS (templ));
 }

 G_OBJECT_CLASS (parent_class)->dispose (object);
}

/* ALWAYS padtemplates cannot have conversion specifications (like src_%d),
 * since it doesn't make sense.
 * SOMETIMES padtemplates can do whatever they want, they are provided by the
 * element.
 * REQUEST padtemplates can be reverse-parsed (the user asks for 'sink1', the
 * 'sink%d' template is automatically selected), so we need to restrict their
 * naming.
 */
static gboolean
name_is_valid (const gchar * name, GstPadPresence presence)
{
 const gchar *str;
```

```c
  if (presence == GST_PAD_ALWAYS) {
    if (strchr (name, '%')) {
      g_warning ("invalid name template %s: conversion specifications are not"
          " allowed for GST_PAD_ALWAYS padtemplates", name);
      return FALSE;
    }
  } else if (presence == GST_PAD_REQUEST) {
    if ((str = strchr (name, '%')) && strchr (str + 1, '%')) {
      g_warning ("invalid name template %s: only one conversion specification"
          " allowed in GST_PAD_REQUEST padtemplate", name);
      return FALSE;
    }
    if (str && (*(str + 1) != 's' && *(str + 1) != 'd' && *(str + 1) != 'u')) {
      g_warning ("invalid name template %s: conversion specification must be of"
          " type '%%d', '%%u' or '%%s' for GST_PAD_REQUEST padtemplate", name);
      return FALSE;
    }
    if (str && (*(str + 2) != '\0')) {
      g_warning ("invalid name template %s: conversion specification must"
          " appear at the end of the GST_PAD_REQUEST padtemplate name", name);
      return FALSE;
    }
  }

  return TRUE;
}

GType
gst_static_pad_template_get_type (void)
{
  static GType staticpadtemplate_type = 0;

  if (G_UNLIKELY (staticpadtemplate_type == 0)) {
    staticpadtemplate_type =
        g_pointer_type_register_static ("GstStaticPadTemplate");
  }
  return staticpadtemplate_type;
}

/**
 * gst_static_pad_template_get:
 * @pad_template: the static pad template
 *
 * Converts a #GstStaticPadTemplate into a #GstPadTemplate.
 *
 * Returns: (transfer full): a new #GstPadTemplate.
 */
/* FIXME0.11: rename to gst_pad_template_new_from_static_pad_template() */
```

```
GstPadTemplate *
gst_static_pad_template_get (GstStaticPadTemplate * pad_template)
{
 GstPadTemplate *new;
 GstCaps *caps;

 if (!name_is_valid (pad_template->name_template, pad_template->presence))
   return NULL;

 caps = gst_static_caps_get (&pad_template->static_caps);

 new = g_object_new (gst_pad_template_get_type (),
    "name", pad_template->name_template,
    "name-template", pad_template->name_template,
    "direction", pad_template->direction,
    "presence", pad_template->presence, "caps", caps, NULL);

 gst_caps_unref (caps);

 return new;
}

/**
* gst_pad_template_new:
* @name_template: the name template.
* @direction: the #GstPadDirection of the template.
* @presence: the #GstPadPresence of the pad.
* @caps: (transfer full): a #GstCaps set for the template. The caps are
*    taken ownership of.
*
* Creates a new pad template with a name according to the given template
* and with the given arguments. This functions takes ownership of the provided
* caps, so be sure to not use them afterwards.
*
* Returns: (transfer full): a new #GstPadTemplate.
*/
GstPadTemplate *
gst_pad_template_new (const gchar * name_template,
   GstPadDirection direction, GstPadPresence presence, GstCaps * caps)
{
 GstPadTemplate *new;

 g_return_val_if_fail (name_template != NULL, NULL);
 g_return_val_if_fail (caps != NULL, NULL);
 g_return_val_if_fail (direction == GST_PAD_SRC
    || direction == GST_PAD_SINK, NULL);
 g_return_val_if_fail (presence == GST_PAD_ALWAYS
    || presence == GST_PAD_SOMETIMES || presence == GST_PAD_REQUEST, NULL);
```

```
if (!name_is_valid (name_template, presence)) {
  gst_caps_unref (caps);
  return NULL;
}

new = g_object_new (gst_pad_template_get_type (),
    "name", name_template, "name-template", name_template,
    "direction", direction, "presence", presence, "caps", caps, NULL);

gst_caps_unref (caps);

return new;
}

/**
* gst_static_pad_template_get_caps:
* @templ: a #GstStaticPadTemplate to get capabilities of.
*
* Gets the capabilities of the static pad template.
*
* Returns: (transfer full): the #GstCaps of the static pad template.
* Unref after usage. Since the core holds an additional
* ref to the returned caps, use gst_caps_make_writable()
* on the returned caps to modify it.
*/
GstCaps *
gst_static_pad_template_get_caps (GstStaticPadTemplate * templ)
{
 g_return_val_if_fail (templ, NULL);

 return (GstCaps *) gst_static_caps_get (&templ->static_caps);
}

/**
* gst_pad_template_get_caps:
* @templ: a #GstPadTemplate to get capabilities of.
*
* Gets the capabilities of the pad template.
*
* Returns: (transfer none): the #GstCaps of the pad template. If you need to
*    keep a reference to the caps, take a ref (see gst_caps_ref ()).
*/
GstCaps *
gst_pad_template_get_caps (GstPadTemplate * templ)
{
 g_return_val_if_fail (GST_IS_PAD_TEMPLATE (templ), NULL);
```

```
 return GST_PAD_TEMPLATE_CAPS (templ);
}

/**
* gst_pad_template_pad_created:
* @templ: a #GstPadTemplate that has been created
* @pad:   the #GstPad that created it
*
* Emit the pad-created signal for this template when created by this pad.
*/
void
gst_pad_template_pad_created (GstPadTemplate * templ, GstPad * pad)
{
 g_signal_emit (templ, gst_pad_template_signals[TEMPL_PAD_CREATED], 0, pad);
}

static void
gst_pad_template_set_property (GObject * object, guint prop_id,
   const GValue * value, GParamSpec * pspec)
{
 /* these properties are all construct-only */
 switch (prop_id) {
  case PROP_NAME_TEMPLATE:
   GST_PAD_TEMPLATE_NAME_TEMPLATE (object) = g_value_dup_string (value);
   break;
  case PROP_DIRECTION:
   GST_PAD_TEMPLATE_DIRECTION (object) =
      (GstPadDirection) g_value_get_enum (value);
   break;
  case PROP_PRESENCE:
   GST_PAD_TEMPLATE_PRESENCE (object) =
      (GstPadPresence) g_value_get_enum (value);
   break;
  case PROP_CAPS:
   /* allow caps == NULL for backwards compatibility (ie. g_object_new()
    * called without any of the new properties) (FIXME 0.11) */
   if (g_value_get_boxed (value) != NULL) {
    GST_PAD_TEMPLATE_CAPS (object) =
       gst_caps_copy (g_value_get_boxed (value));
   }
   break;
  default:
   G_OBJECT_WARN_INVALID_PROPERTY_ID (object, prop_id, pspec);
   break;
 }
}

static void
```

```c
gst_pad_template_get_property (GObject * object, guint prop_id, GValue * value,
    GParamSpec * pspec)
{
 /* these properties are all construct-only */
 switch (prop_id) {
   case PROP_NAME_TEMPLATE:
     g_value_set_string (value, GST_PAD_TEMPLATE_NAME_TEMPLATE (object));
     break;
   case PROP_DIRECTION:
     g_value_set_enum (value, GST_PAD_TEMPLATE_DIRECTION (object));
     break;
   case PROP_PRESENCE:
     g_value_set_enum (value, GST_PAD_TEMPLATE_PRESENCE (object));
     break;
   case PROP_CAPS:
     g_value_set_boxed (value, GST_PAD_TEMPLATE_CAPS (object));
     break;
   default:
     G_OBJECT_WARN_INVALID_PROPERTY_ID (object, prop_id, pspec);
     break;
 }
}
/* GStreamer
* Copyright (C) 1999,2000 Erik Walthinsen <omega@cse.ogi.edu>
*               2000 Wim Taymans <wim.taymans@chello.be>
*
* gstpadtemplate.h: Header for GstPadTemplate object
*
* 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., 59 Temple Place - Suite 330,
* Boston, MA 02111-1307, USA.
*/


#ifndef __GST_PAD_TEMPLATE_H__
#define __GST_PAD_TEMPLATE_H__
```

```
#include <gst/gstconfig.h>

#include <gst/gstobject.h>
#include <gst/gstbuffer.h>
#include <gst/gstcaps.h>
#include <gst/gstevent.h>
#include <gst/gstquery.h>
#include <gst/gsttask.h>

G_BEGIN_DECLS

/* FIXME: this awful circular dependency need to be resolved properly (see pad.h) */
/*typedef struct _GstPadTemplate GstPadTemplate; */
typedef struct _GstPadTemplateClass GstPadTemplateClass;
typedef struct _GstStaticPadTemplate GstStaticPadTemplate;

#define GST_TYPE_STATIC_PAD_TEMPLATE (gst_static_pad_template_get_type ())

#define GST_TYPE_PAD_TEMPLATE  (gst_pad_template_get_type ())
#define GST_PAD_TEMPLATE(obj)  (G_TYPE_CHECK_INSTANCE_CAST ((obj),
GST_TYPE_PAD_TEMPLATE,GstPadTemplate))
#define GST_PAD_TEMPLATE_CLASS(klass) (G_TYPE_CHECK_CLASS_CAST ((klass),
GST_TYPE_PAD_TEMPLATE,GstPadTemplateClass))
#define GST_IS_PAD_TEMPLATE(obj) (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GST_TYPE_PAD_TEMPLATE))
#define GST_IS_PAD_TEMPLATE_CLASS(klass) (G_TYPE_CHECK_CLASS_TYPE ((klass),
GST_TYPE_PAD_TEMPLATE))

/**
* GstPadPresence:
* @GST_PAD_ALWAYS: the pad is always available
* @GST_PAD_SOMETIMES: the pad will become available depending on the media stream
* @GST_PAD_REQUEST: the pad is only available on request with
*  gst_element_get_request_pad().
*
* Indicates when this pad will become available.
*/
typedef enum {
 GST_PAD_ALWAYS,
 GST_PAD_SOMETIMES,
 GST_PAD_REQUEST
} GstPadPresence;

/**
* GST_PAD_TEMPLATE_NAME_TEMPLATE:
* @templ: the template to query
*
* Get the nametemplate of the padtemplate.
```

```
*/
#define GST_PAD_TEMPLATE_NAME_TEMPLATE(templ) (((GstPadTemplate *)(templ))->name_template)


/**
 * GST_PAD_TEMPLATE_DIRECTION:
 * @templ: the template to query
 *
 * Get the #GstPadDirection of the padtemplate.
 */
#define GST_PAD_TEMPLATE_DIRECTION(templ) (((GstPadTemplate *)(templ))->direction)


/**
 * GST_PAD_TEMPLATE_PRESENCE:
 * @templ: the template to query
 *
 * Get the #GstPadPresence of the padtemplate.
 */
#define GST_PAD_TEMPLATE_PRESENCE(templ) (((GstPadTemplate *)(templ))->presence)


/**
 * GST_PAD_TEMPLATE_CAPS:
 * @templ: the template to query
 *
 * Get a handle to the padtemplate #GstCaps
 */
#define GST_PAD_TEMPLATE_CAPS(templ)  (((GstPadTemplate *)(templ))->caps)


/**
 * GstPadTemplateFlags:
 * @GST_PAD_TEMPLATE_FIXED: the padtemplate has no variable properties
 * @GST_PAD_TEMPLATE_FLAG_LAST: first flag that can be used by subclasses.
 *
 * Flags for the padtemplate
 */
typedef enum {
 /* FIXME0.11: this is not used and the purpose is unclear */
 GST_PAD_TEMPLATE_FIXED        = (GST_OBJECT_FLAG_LAST << 0),
 /* padding */
 GST_PAD_TEMPLATE_FLAG_LAST    = (GST_OBJECT_FLAG_LAST << 4)
} GstPadTemplateFlags;


/**
 * GST_PAD_TEMPLATE_IS_FIXED:
 * @templ: the template to query
 *
 * Check if the properties of the padtemplate are fixed
 */
#define GST_PAD_TEMPLATE_IS_FIXED(templ) (GST_OBJECT_FLAG_IS_SET(templ,
```

```
GST_PAD_TEMPLATE_FIXED))

/**
* GstPadTemplate:
*
* The padtemplate object.
*/
struct _GstPadTemplate {
 GstObject    object;

  gchar         *name_template;
 GstPadDirection  direction;
 GstPadPresence   presence;
 GstCaps   *caps;

  gpointer _gst_reserved[GST_PADDING];
};

struct _GstPadTemplateClass {
 GstObjectClass   parent_class;

  /* signal callbacks */
 void (*pad_created) (GstPadTemplate *templ, GstPad *pad);

  gpointer _gst_reserved[GST_PADDING];
};

/**
* GstStaticPadTemplate:
* @name_template: the name of the template
* @direction: the direction of the template
* @presence: the presence of the template
* @static_caps: the caps of the template.
*
* Structure describing the #GstStaticPadTemplate.
*/
struct _GstStaticPadTemplate {
 const gchar     *name_template;
 GstPadDirection  direction;
 GstPadPresence   presence;
 GstStaticCaps    static_caps;
};

/**
* GST_STATIC_PAD_TEMPLATE:
* @padname: the name template of the pad
* @dir: the GstPadDirection of the pad
* @pres: the GstPadPresence of the pad
```

```
 * @caps: the GstStaticCaps of the pad
 *
 * Convenience macro to fill the values of a GstStaticPadTemplate
 * structure.
 */
#define GST_STATIC_PAD_TEMPLATE(padname, dir, pres, caps) \
{ \
 /* name_template */   padname, \
 /* direction */       dir, \
 /* presence */        pres, \
 /* caps */            caps \
}


/* templates and factories */
GType   gst_pad_template_get_type  (void);
GType   gst_static_pad_template_get_type (void);


GstPadTemplate*  gst_pad_template_new   (const gchar *name_template,
      GstPadDirection direction, GstPadPresence presence,
      GstCaps *caps) G_GNUC_MALLOC;


GstPadTemplate * gst_static_pad_template_get        (GstStaticPadTemplate *pad_template);
GstCaps*  gst_static_pad_template_get_caps (GstStaticPadTemplate *templ);
GstCaps*  gst_pad_template_get_caps  (GstPadTemplate *templ);


void             gst_pad_template_pad_created        (GstPadTemplate * templ, GstPad * pad);


G_END_DECLS


#endif /* __GST_PAD_TEMPLATE_H__ */
```

```
*/

#ifndef CHECK_IMPL_H
#define CHECK_IMPL_H


/* This header should be included by any module that needs
   to know the implementation details of the check structures
   Include stdio.h & list.h before this header
*/

typedef struct TF {
 TFun fn;
 int loop_start;
 int loop_end;
 const char *name;
 int signal;
 unsigned char allowed_exit_value;
} TF;

struct Suite {
 const char *name;
 List *tclst; /* List of test cases */
};

typedef struct Fixture
{
 int ischecked;
 SFun fun;
} Fixture;

struct TCase {
 const char *name;
 int timeout;
 List *tflst; /* list of test functions */
 List *unch_sflst;
 List *unch_tflst;
 List *ch_sflst;
 List *ch_tflst;
};

typedef struct TestStats {
 int n_checked;
 int n_failed;
 int n_errors;
} TestStats;

struct TestResult {
```

```c
  enum test_result rtype;    /* Type of result */
  enum ck_result_ctx ctx;    /* When the result occurred */
  char *file;   /* File where the test occured */
  int line;     /* Line number where the test occurred */
  int iter;     /* The iteration value for looping tests */
  const char *tcname;  /* Test case that generated the result */
  const char *tname;  /* Test that generated the result */
  char *msg;    /* Failure message */
};

TestResult *tr_create(void);
void tr_reset(TestResult *tr);

enum cl_event {
 CLINITLOG_SR,
 CLENDLOG_SR,
 CLSTART_SR,
 CLSTART_S,
 CLEND_SR,
 CLEND_S,
 CLSTART_T, /* A test case is about to run */
 CLEND_T
};

typedef void (*LFun) (SRunner *, FILE*, enum print_output,
       void *, enum cl_event);

typedef struct Log {
 FILE *lfile;
 LFun lfun;
 int close;
 enum print_output mode;
} Log;

struct SRunner {
 List *slst; /* List of Suite objects */
 TestStats *stats; /* Run statistics */
 List *resultlst; /* List of unit test results */
 const char *log_fname; /* name of log file */
 const char *xml_fname; /* name of xml output file */
 List *loglst; /* list of Log objects */
 enum fork_status fstat; /* controls if suites are forked or not
       NOTE: Don't use this value directly,
       instead use srunner_fork_status */
};

void set_fork_status(enum fork_status fstat);
```

```c
enum fork_status cur_fork_status (void);

#endif /* CHECK_IMPL_H */
/* GStreamer
 * Copyright (C) 1999,2000 Erik Walthinsen <omega@cse.ogi.edu>
 *               2000 Wim Taymans <wtay@chello.be>
 *               2005 Wim Taymans <wim@fluendo.com>
 *               2005 David Schleef <ds@schleef.org>
 *
 * 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., 59 Temple Place - Suite 330,
 * Boston, MA 02111-1307, USA.
 */
/**
 * SECTION:element-capsfilter
 *
 * The element does not modify data as such, but can enforce limitations on the
 * data format.
 *
 * <refsect2>
 * <title>Example launch line</title>
 * |[
 * gst-launch videotestsrc ! video/x-raw-gray ! ffmpegcolorspace ! autovideosink
 * ]| Limits acceptable video from videotestsrc to be grayscale.
 * </refsect2>
 */

#ifdef HAVE_CONFIG_H
#include "config.h"
#endif

#include "../../gst/gst-i18n-lib.h"
#include "gstcapsfilter.h"

enum
{
  PROP_0,
```

```
 PROP_FILTER_CAPS
};


static GstStaticPadTemplate sinktemplate = GST_STATIC_PAD_TEMPLATE ("sink",
  GST_PAD_SINK,
  GST_PAD_ALWAYS,
  GST_STATIC_CAPS_ANY);

static GstStaticPadTemplate srctemplate = GST_STATIC_PAD_TEMPLATE ("src",
  GST_PAD_SRC,
  GST_PAD_ALWAYS,
  GST_STATIC_CAPS_ANY);


GST_DEBUG_CATEGORY_STATIC (gst_capsfilter_debug);
#define GST_CAT_DEFAULT gst_capsfilter_debug

#define _do_init(bla) \
  GST_DEBUG_CATEGORY_INIT (gst_capsfilter_debug, "capsfilter", 0, \
  "capsfilter element");

GST_BOILERPLATE_FULL (GstCapsFilter, gst_capsfilter, GstBaseTransform,
  GST_TYPE_BASE_TRANSFORM, _do_init);


static void gst_capsfilter_set_property (GObject * object, guint prop_id,
  const GValue * value, GParamSpec * pspec);
static void gst_capsfilter_get_property (GObject * object, guint prop_id,
  GValue * value, GParamSpec * pspec);
static void gst_capsfilter_dispose (GObject * object);

static GstCaps *gst_capsfilter_transform_caps (GstBaseTransform * base,
  GstPadDirection direction, GstCaps * caps);
static gboolean gst_capsfilter_accept_caps (GstBaseTransform * base,
  GstPadDirection direction, GstCaps * caps);
static GstFlowReturn gst_capsfilter_transform_ip (GstBaseTransform * base,
  GstBuffer * buf);
static GstFlowReturn gst_capsfilter_prepare_buf (GstBaseTransform * trans,
  GstBuffer * input, gint size, GstCaps * caps, GstBuffer ** buf);

static void
gst_capsfilter_base_init (gpointer g_class)
{
GstElementClass *gstelement_class = GST_ELEMENT_CLASS (g_class);

 gst_element_class_set_details_simple (gstelement_class,
    "CapsFilter",
```

```c
    "Generic",
    "Pass data without modification, limiting formats",
    "David Schleef <ds@schleef.org>");
  gst_element_class_add_pad_template (gstelement_class,
    gst_static_pad_template_get (&srctemplate));
  gst_element_class_add_pad_template (gstelement_class,
    gst_static_pad_template_get (&sinktemplate));
}

static void
gst_capsfilter_class_init (GstCapsFilterClass * klass)
{
 GObjectClass *gobject_class;
 GstBaseTransformClass *trans_class;

 gobject_class = G_OBJECT_CLASS (klass);
 gobject_class->set_property = gst_capsfilter_set_property;
 gobject_class->get_property = gst_capsfilter_get_property;
 gobject_class->dispose = gst_capsfilter_dispose;

 g_object_class_install_property (gobject_class, PROP_FILTER_CAPS,
    g_param_spec_boxed ("caps", _("Filter caps"),
      _("Restrict the possible allowed capabilities (NULL means ANY). "
        "Setting this property takes a reference to the supplied GstCaps "
        "object."), GST_TYPE_CAPS,
      G_PARAM_READWRITE | G_PARAM_STATIC_STRINGS));

 trans_class = GST_BASE_TRANSFORM_CLASS (klass);
 trans_class->transform_caps =
    GST_DEBUG_FUNCPTR (gst_capsfilter_transform_caps);
 trans_class->transform_ip = GST_DEBUG_FUNCPTR (gst_capsfilter_transform_ip);
 trans_class->accept_caps = GST_DEBUG_FUNCPTR (gst_capsfilter_accept_caps);
 trans_class->prepare_output_buffer =
    GST_DEBUG_FUNCPTR (gst_capsfilter_prepare_buf);
}

static void
gst_capsfilter_init (GstCapsFilter * filter, GstCapsFilterClass * g_class)
{
 GstBaseTransform *trans = GST_BASE_TRANSFORM (filter);
 gst_base_transform_set_gap_aware (trans, TRUE);
 filter->filter_caps = gst_caps_new_any ();
}

static gboolean
copy_func (GQuark field_id, const GValue * value, GstStructure * dest)
{
 gst_structure_id_set_value (dest, field_id, value);
```

```c
 return TRUE;
}

static void
gst_capsfilter_set_property (GObject * object, guint prop_id,
  const GValue * value, GParamSpec * pspec)
{
 GstCapsFilter *capsfilter = GST_CAPSFILTER (object);

 switch (prop_id) {
  case PROP_FILTER_CAPS:{
    GstCaps *new_caps;
    GstCaps *old_caps, *suggest, *nego;
    const GstCaps *new_caps_val = gst_value_get_caps (value);

    if (new_caps_val == NULL) {
     new_caps = gst_caps_new_any ();
    } else {
     new_caps = (GstCaps *) new_caps_val;
     gst_caps_ref (new_caps);
    }

    GST_OBJECT_LOCK (capsfilter);
    old_caps = capsfilter->filter_caps;
    capsfilter->filter_caps = new_caps;
    GST_OBJECT_UNLOCK (capsfilter);

    gst_caps_unref (old_caps);

    GST_DEBUG_OBJECT (capsfilter, "set new caps %" GST_PTR_FORMAT, new_caps);

    /* filter the currently negotiated format against the new caps */
    GST_OBJECT_LOCK (GST_BASE_TRANSFORM_SINK_PAD (object));
    nego = GST_PAD_CAPS (GST_BASE_TRANSFORM_SINK_PAD (object));
    if (nego) {
     GST_DEBUG_OBJECT (capsfilter, "we had negotiated caps %" GST_PTR_FORMAT,
        nego);

     if (G_UNLIKELY (gst_caps_is_any (new_caps))) {
       GST_DEBUG_OBJECT (capsfilter, "not settings any suggestion");

       suggest = NULL;
     } else {
       GstStructure *s1, *s2;

       /* first check if the name is the same */
       s1 = gst_caps_get_structure (nego, 0);
```

```
      s2 = gst_caps_get_structure (new_caps, 0);

      if (gst_structure_get_name_id (s1) == gst_structure_get_name_id (s2)) {
       /* same name, copy all fields from the new caps into the previously
        * negotiated caps */
       suggest = gst_caps_copy (nego);
       s1 = gst_caps_get_structure (suggest, 0);
       gst_structure_foreach (s2, (GstStructureForeachFunc) copy_func, s1);
       GST_DEBUG_OBJECT (capsfilter, "copied structure fields");
      } else {
       GST_DEBUG_OBJECT (capsfilter, "different structure names");
       /* different names, we can only suggest the complete caps */
       suggest = gst_caps_copy (new_caps);
      }
     }
    } else {
     GST_DEBUG_OBJECT (capsfilter, "no negotiated caps");
     /* Suggest the new caps, we can't just rely on _get_caps as this may
      * already be called at this point even though no buffer has been
      * pushed yet */
     suggest = gst_caps_copy (new_caps);
    }
    GST_OBJECT_UNLOCK (GST_BASE_TRANSFORM_SINK_PAD (object));

    GST_DEBUG_OBJECT (capsfilter, "suggesting new caps %" GST_PTR_FORMAT,
        suggest);
    gst_base_transform_suggest (GST_BASE_TRANSFORM (object), suggest, 0);
    if (suggest)
     gst_caps_unref (suggest);

    break;
   }
  default:
   G_OBJECT_WARN_INVALID_PROPERTY_ID (object, prop_id, pspec);
   break;
 }
}

static void
gst_capsfilter_get_property (GObject * object, guint prop_id, GValue * value,
  GParamSpec * pspec)
{
 GstCapsFilter *capsfilter = GST_CAPSFILTER (object);

 switch (prop_id) {
  case PROP_FILTER_CAPS:
   GST_OBJECT_LOCK (capsfilter);
   gst_value_set_caps (value, capsfilter->filter_caps);
```

```
    GST_OBJECT_UNLOCK (capsfilter);
    break;
  default:
    G_OBJECT_WARN_INVALID_PROPERTY_ID (object, prop_id, pspec);
    break;
 }
}

static void
gst_capsfilter_dispose (GObject * object)
{
 GstCapsFilter *filter = GST_CAPSFILTER (object);

 gst_caps_replace (&filter->filter_caps, NULL);

 G_OBJECT_CLASS (parent_class)->dispose (object);
}

static GstCaps *
gst_capsfilter_transform_caps (GstBaseTransform * base,
  GstPadDirection direction, GstCaps * caps)
{
 GstCapsFilter *capsfilter = GST_CAPSFILTER (base);
 GstCaps *ret, *filter_caps;

 GST_OBJECT_LOCK (capsfilter);
 filter_caps = gst_caps_ref (capsfilter->filter_caps);
 GST_OBJECT_UNLOCK (capsfilter);

 ret = gst_caps_intersect (caps, filter_caps);
 GST_DEBUG_OBJECT (capsfilter, "input:    %" GST_PTR_FORMAT, caps);
 GST_DEBUG_OBJECT (capsfilter, "filter:   %" GST_PTR_FORMAT, filter_caps);
 GST_DEBUG_OBJECT (capsfilter, "intersect: %" GST_PTR_FORMAT, ret);

 gst_caps_unref (filter_caps);

 return ret;
}

static gboolean
gst_capsfilter_accept_caps (GstBaseTransform * base,
  GstPadDirection direction, GstCaps * caps)
{
 GstCapsFilter *capsfilter = GST_CAPSFILTER (base);
 GstCaps *filter_caps;
 gboolean ret;

 GST_OBJECT_LOCK (capsfilter);
```

```
  filter_caps = gst_caps_ref (capsfilter->filter_caps);
  GST_OBJECT_UNLOCK (capsfilter);

  ret = gst_caps_can_intersect (caps, filter_caps);
  GST_DEBUG_OBJECT (capsfilter, "can intersect: %d", ret);
  if (ret) {
    /* if we can intersect, see if the other end also accepts */
    if (direction == GST_PAD_SRC)
      ret = gst_pad_peer_accept_caps (GST_BASE_TRANSFORM_SINK_PAD (base), caps);
    else
      ret = gst_pad_peer_accept_caps (GST_BASE_TRANSFORM_SRC_PAD (base), caps);
    GST_DEBUG_OBJECT (capsfilter, "peer accept: %d", ret);
  }

  gst_caps_unref (filter_caps);

  return ret;
}

static GstFlowReturn
gst_capsfilter_transform_ip (GstBaseTransform * base, GstBuffer * buf)
{
  /* No actual work here. It's all done in the prepare output buffer
   * func. */
  return GST_FLOW_OK;
}

/* Output buffer preparation... if the buffer has no caps, and
 * our allowed output caps is fixed, then give the caps to the
 * buffer.
 * This ensures that outgoing buffers have caps if we can, so
 * that pipelines like:
 *   gst-launch filesrc location=rawsamples.raw !
 *     audio/x-raw-int,width=16,depth=16,rate=48000,channels=2,
 *     endianness=4321,signed='(boolean)'true ! alsasink
 * will work.
 */
static GstFlowReturn
gst_capsfilter_prepare_buf (GstBaseTransform * trans, GstBuffer * input,
    gint size, GstCaps * caps, GstBuffer ** buf)
{
  GstFlowReturn ret = GST_FLOW_OK;

  if (GST_BUFFER_CAPS (input) != NULL) {
    /* Output buffer already has caps */
    GST_LOG_OBJECT (trans, "Input buffer already has caps (implicitly fixed)");
    /* FIXME : Move this behaviour to basetransform. The given caps are the ones
     * of the source pad, therefore our outgoing buffers should always have
```

```
 * those caps. */
if (GST_BUFFER_CAPS (input) != caps) {
 /* caps are different, make a metadata writable output buffer to set
  * caps */
 if (gst_buffer_is_metadata_writable (input)) {
  /* input is writable, just set caps and use this as the output */
  *buf = input;
  gst_buffer_set_caps (*buf, caps);
  gst_buffer_ref (input);
 } else {
  GST_DEBUG_OBJECT (trans, "Creating sub-buffer and setting caps");
  *buf = gst_buffer_create_sub (input, 0, GST_BUFFER_SIZE (input));
  gst_buffer_set_caps (*buf, caps);
 }
} else {
 /* caps are right, just use a ref of the input as the outbuf */
 *buf = input;
 gst_buffer_ref (input);
}
} else {
/* Buffer has no caps. See if the output pad only supports fixed caps */
GstCaps *out_caps;

out_caps = GST_PAD_CAPS (trans->srcpad);

if (out_caps != NULL) {
 gst_caps_ref (out_caps);
} else {
 out_caps = gst_pad_get_allowed_caps (trans->srcpad);
 g_return_val_if_fail (out_caps != NULL, GST_FLOW_ERROR);
}

out_caps = gst_caps_make_writable (out_caps);
gst_caps_do_simplify (out_caps);

if (gst_caps_is_fixed (out_caps) && !gst_caps_is_empty (out_caps)) {
 GST_DEBUG_OBJECT (trans, "Have fixed output caps %"
    GST_PTR_FORMAT " to apply to buffer with no caps", out_caps);
 if (gst_buffer_is_metadata_writable (input)) {
  gst_buffer_ref (input);
  *buf = input;
 } else {
  GST_DEBUG_OBJECT (trans, "Creating sub-buffer and setting caps");
  *buf = gst_buffer_create_sub (input, 0, GST_BUFFER_SIZE (input));
 }
 GST_BUFFER_CAPS (*buf) = out_caps;

 if (GST_PAD_CAPS (trans->srcpad) == NULL)
```

```
    gst_pad_set_caps (trans->srcpad, out_caps);
  } else {
    gchar *caps_str = gst_caps_to_string (out_caps);

    GST_DEBUG_OBJECT (trans, "Cannot choose caps. Have unfixed output caps %"
      GST_PTR_FORMAT, out_caps);
    gst_caps_unref (out_caps);

    ret = GST_FLOW_ERROR;
    GST_ELEMENT_ERROR (trans, STREAM, FORMAT,
      ("Filter caps do not completely specify the output format"),
      ("Output caps are unfixed: %s", caps_str));
    g_free (caps_str);
  }
 }

 return ret;
}
/* GStreamer
 * Copyright (C) 1999,2000 Erik Walthinsen <omega@cse.ogi.edu>
 *               2000 Wim Taymans <wtay@chello.be>
 *
 * gstidentity.h:
 *
 * 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., 59 Temple Place - Suite 330,
 * Boston, MA 02111-1307, USA.
 */



#ifndef __GST_CAPSFILTER_H__
#define __GST_CAPSFILTER_H__



#include <gst/gst.h>
#include <gst/base/gstbasetransform.h>
```

```
G_BEGIN_DECLS

#define GST_TYPE_CAPSFILTER \
 (gst_capsfilter_get_type())
#define GST_CAPSFILTER(obj) \
 (G_TYPE_CHECK_INSTANCE_CAST((obj),GST_TYPE_CAPSFILTER,GstCapsFilter))
#define GST_CAPSFILTER_CLASS(klass) \
 (G_TYPE_CHECK_CLASS_CAST((klass),GST_TYPE_CAPSFILTER,GstCapsFilterClass))
#define GST_IS_CAPSFILTER(obj) \
 (G_TYPE_CHECK_INSTANCE_TYPE((obj),GST_TYPE_CAPSFILTER))
#define GST_IS_CAPSFILTER_CLASS(klass) \
 (G_TYPE_CHECK_CLASS_TYPE((klass),GST_TYPE_CAPSFILTER))

typedef struct _GstCapsFilter GstCapsFilter;
typedef struct _GstCapsFilterClass GstCapsFilterClass;

/**
* GstCapsFilter:
*
* The opaque #GstCapsFilter data structure.
*/
struct _GstCapsFilter {
 GstBaseTransform trans;

 GstCaps *filter_caps;
};

struct _GstCapsFilterClass {
 GstBaseTransformClass trans_class;
};

GType gst_capsfilter_get_type (void);

G_END_DECLS

#endif /* __GST_CAPSFILTER_H__ */
```

1.6 gstreamer-plugins-bad 0.10.22.1

1.6.1 Available under license :

GNU GENERAL PUBLIC LICENSE
     Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
          59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Library General Public License instead.) You can apply it to
your programs, 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 software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, 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 redistributors of a free
program will individually obtain patent licenses, in effect making the
program proprietary. To prevent this, we have made it clear that any
patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

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

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's 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 give any other recipients of the Program a copy of this License along with the Program.

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 Program or any portion of it, thus forming a work based on the Program, 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) You must cause the modified files to carry prominent notices
  stating that you changed the files and the date of any change.

  b) You must cause any work that you distribute or publish, that in
  whole or in part contains or is derived from the Program or any
  part thereof, to be licensed as a whole at no charge to all third
  parties under the terms of this License.

  c) If the modified program normally reads commands interactively

when run, you must cause it, when started running for such
interactive use in the most ordinary way, to print or display an
announcement including an appropriate copyright notice and a
notice that there is no warranty (or else, saying that you provide
a warranty) and that users may redistribute the program under
these conditions, and telling the user how to view a copy of this
License. (Exception: if the Program itself is interactive but
does not normally print such an announcement, your work based on
the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If
identifiable sections of that work are not derived from the Program,
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 Program, 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 Program.

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

 3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of
Sections 1 and 2 above provided that you also do one of the following:

  a) 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; or,

  b) Accompany it with a written offer, valid for at least three
  years, to give any third party, for a charge no more than your
  cost of physically performing source distribution, a complete
  machine-readable copy of the corresponding source code, to be
  distributed under the terms of Sections 1 and 2 above on a medium
  customarily used for software interchange; or,

  c) Accompany it with the information you received as to the offer
  to distribute corresponding source code. (This alternative is
  allowed only for noncommercial distribution and only if you

received the program in object code or executable form with such
an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, 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 executable. 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.

If distribution of executable or 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 counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

 4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense or distribute the Program 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.

 5. 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 Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

 6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program 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.

 7. 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 Program at all. For example, if a patent
license would not permit royalty-free redistribution of the Program 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 Program.

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.

 8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program 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.

 9. The Free Software Foundation may publish revised and/or new versions
of the 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 Program
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 Program does not specify a version number of
this License, you may choose any version ever published by the Free Software

Foundation.

10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, 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

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "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 PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. 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 PROGRAM 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 PROGRAM (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 PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), 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 Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. 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 program's name and a brief idea of what it does.>
Copyright (C) 19yy <name of author>

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

This program 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 General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

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

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) 19yy name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

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

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Library General
Public License instead of this License.

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!
/* GStreamer
 * Copyright (C) <1999> Erik Walthinsen <omega@cse.ogi.edu>

```
*
* 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., 59 Temple Place - Suite 330,
* Boston, MA 02111-1307, USA.
*/
```

Disclaimer: The GStreamer team makes a real effort to make sure that
the information in this file is as up-to-date and accurate as possible.
However, this file may contain errors and omissions. Some dependant
libraries change their licensing from version-to-version and some of
the libraries listed here as LGPL were under the GPL license in
older releases. Distributions which include GStreamer are responsible
for ensuring that plugins are used in accordance with licensing terms
and other laws.

GStreamer is developed under the terms of the LGPL (see LICENSE file for details).
Some of our plugins however rely on libraries which are available under other
licenses. This means that if you are using an application which has a non-GPL
compatible license, for instance a closed-source application with GStreamer
you have to make sure not to use GPL linked or derived plugins. When using
GPL linked plugins GStreamer is for all practical reasons under the GPL itself.

The plugins which use a GPL library are as follows:

```
a52dec        liba52        (http://liba52.sourceforge.net/)
aasink        aalib         (http://sourceforge.net/projects/aa-project/)
cdparanoia    libcdparanoia  (http://www.xiph.org/paranoia/)
dtsdec        libdts        (http://www.videolan.org/dtsdec.html)
dvdnavsrc     libdvdnav     (http://dvd.sourceforge.net/)
dxr3                        (http://dxr3.sourceforge.net/)
icecastsend   libshout      (http://www.icecast.org)
mad           libmad        (http://www.mars.org/home/rob/proj/mpeg/)
mpeg2dec      libmpeg2      (http://libmpeg2.sourceforge.net/)
mpeg2enc      libmpeg2enc   (http://mjpeg.sourceforge.net/)
mplex         libmplex      (http://mjpeg.sourceforge.net/)
siddec        libsidplay 1.36 (http://www.geocities.com/SiliconValley/Lakes/5147/sidplay/,
                             http://sourceforge.net/projects/sidplay2/)
```

trm   libmusicbrainz   (http://www.musicbrainz.org/)

xine   libxine   (http://xinehq.de/)

xvid   libxvidcore   (http://www.xvid.org/)

Plugins derived from GPL code are as follows:

dvdreadsrc   libdvdread   (http://www.dtek.chalmers.se/groups/dvd/)

monoscope   None   (Algorithm by Ralph Loader, Joerg Walter,

          Richard Boulton, and Andy Lo A Foe)

rtjpeg   None   (Erik Walthinsen's algorithm)

rtp   None   (http://www.linphone.org/ortp/)

synaesthesia   (http://www.logarithmic.net/pfh/synaesthesia)

system_encode   None   (Algorithm by Chrisoph Moar, Wim Tayman's and

          Erik Walthinsen)

vbidec   None   (Algorithm by Billy Biggs, Doug Bell,

          Erik Walthinsen and David I. Lehn)

Plugins which use a LGPL library are as follows:

alsa   alsa   (http://alsa-project.org/)

artsdsink   aRts   (http://arts-project.org/)

cacasink   libcaca   (http://sam.zoy.org/projects/libcaca/)

colorspacelcs   libcolorspace

dvdec   libdv   (http://libdv.sourceforge.net/)

esdmon,esdsink   libesd   (ftp.gnome.org/pub/GNOME/stable/sources/esound/)

faac   libfaac   (http://www.audiocoding.com/modules/mydownloads/)

fameenc   libfame   (http://fame.sourceforge.net/)

ffmpeg   ffmpeg   (http://ffmpeg.sourceforge.net/)

gdkpixbuf   GTK+   (http://www.gtk.org/)

gnomevfs   gnome-vfs   (ftp.gnome.org//pub/GNOME/stable/sources/gnome-vfs/)

gst_arts   aRts   (http://arts-project.org/)

gst1394   libraw1394   (http://www.linux1394.org/)

gstaf   libaudiofile   (http://www.68k.org/~michael/audiofile/)

gstsf   libsndfile   (http://www.mega-nerd.com/libsndfile/)

hermescolorspace Hermes   (http://www.clanlib.org/hermes/)

kio   KDE   (http://www.kde.org/)

ladspa   (http://www.ladspa.org/)

lame   libmp3lame   (http://lame.sourceforge.net/)

libvisual   libvisual   (http://libvisual.sourceforge.net/)

matroska   (http://www.matroska.org/)

mikmod   libmikmod   (http://mikmod.raphnet.net/)

ossaudio   (http://www.opensound.com/)

qcamsrc

rfbenc   librfb

sdlvideosink   libsdl   (http://www.libsdl.org/)

shout2send   libshout2   (http://www.icecast.org)

smoothwave

swfdec   (http://swfdec.sourceforge.net/)

tarkin   (http://svn.xiph.org/trunk/tarkin/)

textoverlay   pango   (http://www.pango.org/)

dirac   (http://www.bbc.co.uk/rd/projects/dirac/)

effectv (Our ports was relicensed)(http://effectv.sourceforge.net/)

musepack   (http://www.musepack.net/)

kate   libtiger   (http://libtiger.googlecode.com/)
                         Optional, only for the tiger element

Plugins which use a BSD covered library are as follows:

ogg   libogg/libvorbis (http://www.xiph.org/ogg/vorbis/)

vorbis   libogg/libvorbis (http://www.xiph.org/ogg/vorbis/)

gsttheora   libtheora   (http://www.theora.org/)

speex   (http://www.speex.org/)

flac   libFLAC   (http://flac.sourceforge.net/)

kate   libkate   (http://libkate.googlecode.com/)

Plugins based on libraries with other free licenses:

ximagesink   libXv

xvimagesink   libXv

- license: MIT X11 / X Consortium license

gsm   libgsm

- license   MIT license http://kbs.cs.tu-berlin.de/~jutta/toast.html

festival   (http://www.cstr.ed.ac.uk/projects/festival/)

- license: http://www.cstr.ed.ac.uk/projects/festival/freecopyright.shtml

jpeg   (http://www.ijg.org/)

- license: IJG license

nas   (http://radscan.com/nas.html)

- license: NAS license

snapshot   libpng   (http://www.libpng.org/pub/png/)

- license: PNG license

mngdec/mngenc libmng   (http://gjuyn.xs4all.nl/libmng)

- license   PNG license

Plugins using non-free libraries:

divxdec, divxenc   (http://www.divx.com/)

osxaudio   (http://www.apple.com/macosx/)

sunaudiosink   (http://www.sun.com/)

Unsure:

faad   libfaad   (http://www.audiocoding.com/)

faad's license depends on its major version :

 - for FAAD1 : it is LGPL

 - for FAAD2 : it is GPL

look at these pages for more information on licenses :

http://www.audiocoding.com/modules/wiki/?page=FAAD2

http://www.audiocoding.com/modules/wiki/?page=FAAC

by default, GStreamer faad plugin would compile against FAAD2 if available.

it falls back to FAAD1 if not.

/* GStreamer

* Copyright (C) <1999> Erik Walthinsen <omega@cse.ogi.edu>

* Copyright (C) 2003,2004 David A. Schleef <ds@schleef.org>

*

* 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., 59 Temple Place - Suite 330,

* Boston, MA 02111-1307, USA.

*/

/* Element-Checklist-Version: 5 */

#ifdef HAVE_CONFIG_H

#include "config.h"

#endif

#include <string.h>

#include <math.h>

/*#define DEBUG_ENABLED */

#include "gstaudioresample.h"

#include <gst/audio/audio.h>

GST_DEBUG_CATEGORY_STATIC (audioresample_debug);

#define GST_CAT_DEFAULT audioresample_debug

/* Audioresample signals and args */

enum

{

```
/* FILL ME */
LAST_SIGNAL
};

enum
{
ARG_0,
ARG_FILTERLEN
};

#define SUPPORTED_CAPS \
 GST_STATIC_CAPS (\
   "audio/x-raw-int, " \
     "rate = (int) [ 1, MAX ], " \
     "channels = (int) [ 1, MAX ], " \
     "endianness = (int) BYTE_ORDER, " \
     "width = (int) 16, " \
     "depth = (int) 16, " \
     "signed = (boolean) true"
#if 0
   /* disabled because it segfaults */
"audio/x-raw-float, "
  "rate = (int) [ 1, MAX ], "
  "channels = (int) [ 1, MAX ], "
  "endianness = (int) BYTE_ORDER, " "width = (int) 32"
#endif
   )

 static GstStaticPadTemplate gst_audioresample_sink_template =
     GST_STATIC_PAD_TEMPLATE ("sink",
     GST_PAD_SINK, GST_PAD_ALWAYS, SUPPORTED_CAPS);

 static GstStaticPadTemplate gst_audioresample_src_template =
     GST_STATIC_PAD_TEMPLATE ("src",
     GST_PAD_SRC, GST_PAD_ALWAYS, SUPPORTED_CAPS);

 static void gst_audioresample_base_init (gpointer g_class);
 static void gst_audioresample_class_init (AudioresampleClass * klass);
 static void gst_audioresample_init (Audioresample * audioresample);
 static void gst_audioresample_dispose (GObject * object);

 static void gst_audioresample_chain (GstPad * pad, GstData * _data);

 static void gst_audioresample_set_property (GObject * object,
     guint prop_id, const GValue * value, GParamSpec * pspec);
 static void gst_audioresample_get_property (GObject * object,
     guint prop_id, GValue * value, GParamSpec * pspec);
```

```c
static GstElementClass *parent_class = NULL;

/*static guint gst_audioresample_signals[LAST_SIGNAL] = { 0 }; */

GType audioresample_get_type (void)
{
  static GType audioresample_type = 0;

  if (!audioresample_type)
  {
    static const GTypeInfo audioresample_info = {
    sizeof (AudioresampleClass),
        gst_audioresample_base_init,
        NULL,
        (GClassInitFunc) gst_audioresample_class_init,
        NULL,
        NULL,
        sizeof (Audioresample), 0,
        (GInstanceInitFunc) gst_audioresample_init,};

    audioresample_type =
        g_type_register_static (GST_TYPE_ELEMENT, "Audioresample",
        &audioresample_info, 0);
  }
  return audioresample_type;
}

static void gst_audioresample_base_init (gpointer g_class)
{
 GstElementClass *gstelement_class = GST_ELEMENT_CLASS (g_class);

 gst_element_class_add_pad_template (gstelement_class,
    gst_static_pad_template_get (&gst_audioresample_src_template));
 gst_element_class_add_pad_template (gstelement_class,
    gst_static_pad_template_get (&gst_audioresample_sink_template));

 gst_element_class_set_details_simple (gstelement_class, "Audio scaler",
    "Filter/Converter/Audio",
    "Resample audio", "David Schleef <ds@schleef.org>");
}

static void gst_audioresample_class_init (AudioresampleClass * klass)
{
 GObjectClass *gobject_class;
 GstElementClass *gstelement_class;

 gobject_class = (GObjectClass *) klass;
 gstelement_class = (GstElementClass *) klass;
```

```c
gobject_class->set_property = gst_audioresample_set_property;
gobject_class->get_property = gst_audioresample_get_property;
gobject_class->dispose = gst_audioresample_dispose;

g_object_class_install_property (G_OBJECT_CLASS (klass), ARG_FILTERLEN,
    g_param_spec_int ("filter-length", "filter_length", "filter_length",
        0, G_MAXINT, 16,
        G_PARAM_READWRITE | G_PARAM_CONSTRUCT | G_PARAM_STATIC_STRINGS));

parent_class = g_type_class_peek_parent (klass);

GST_DEBUG_CATEGORY_INIT (audioresample_debug, "audioresample", 0,
    "audioresample element");
}

static void gst_audioresample_expand_caps (GstCaps * caps)
{
 gint i;

 for (i = 0; i < gst_caps_get_size (caps); i++) {
   GstStructure *structure = gst_caps_get_structure (caps, i);
   const GValue *value;

   value = gst_structure_get_value (structure, "rate");
   if (value == NULL) {
     GST_ERROR ("caps structure doesn't have required rate field");
     return;
   }

   gst_structure_set (structure, "rate", GST_TYPE_INT_RANGE, 1, G_MAXINT, 0);
 }
}

static GstCaps *gst_audioresample_getcaps (GstPad * pad)
{
 Audioresample *audioresample;
 GstCaps *caps;
 GstPad *otherpad;

 audioresample = GST_AUDIORESAMPLE (gst_pad_get_parent (pad));

 otherpad = (pad == audioresample->srcpad) ? audioresample->sinkpad :
    audioresample->srcpad;
 caps = gst_pad_get_allowed_caps (otherpad);

 gst_audioresample_expand_caps (caps);
```

```c
  return caps;
}

static GstCaps *gst_audioresample_fixate (GstPad * pad, const GstCaps * caps)
{
 Audioresample *audioresample;
 GstPad *otherpad;
 int rate;
 GstCaps *copy;
 GstStructure *structure;

   audioresample = GST_AUDIORESAMPLE (gst_pad_get_parent (pad));

 if (pad == audioresample->srcpad) {
   otherpad = audioresample->sinkpad;
   rate = audioresample->i_rate;
 } else
 {
   otherpad = audioresample->srcpad;
   rate = audioresample->o_rate;
 }
 if (!GST_PAD_IS_NEGOTIATING (otherpad))
   return NULL;
 if (gst_caps_get_size (caps) > 1)
   return NULL;

 copy = gst_caps_copy (caps);
 structure = gst_caps_get_structure (copy, 0);
 if (rate) {
   if (gst_structure_fixate_field_nearest_int (structure, "rate", rate)) {
     return copy;
   }
 }
 gst_caps_free (copy);
 return NULL;
}

static GstPadLinkReturn gst_audioresample_link (GstPad * pad,
   const GstCaps * caps)
{
 Audioresample *audioresample;
 GstStructure *structure;
 int rate;
 int channels;
 gboolean ret;
 GstPad *otherpad;

   audioresample = GST_AUDIORESAMPLE (gst_pad_get_parent (pad));
```

```
  otherpad = (pad == audioresample->srcpad) ? audioresample->sinkpad :
    audioresample->srcpad;

  structure = gst_caps_get_structure (caps, 0);
  ret = gst_structure_get_int (structure, "rate", &rate);
  ret &= gst_structure_get_int (structure, "channels", &channels);
if (!ret)
{
  return GST_PAD_LINK_REFUSED;
}

if (gst_pad_is_negotiated (otherpad))
{
  GstCaps *othercaps = gst_caps_copy (caps);
  int otherrate;
  GstPadLinkReturn linkret;

  if (pad == audioresample->srcpad) {
    otherrate = audioresample->i_rate;
  } else {
    otherrate = audioresample->o_rate;
  }
  gst_caps_set_simple (othercaps, "rate", G_TYPE_INT, otherrate, NULL);
  linkret = gst_pad_try_set_caps (otherpad, othercaps);
  if (GST_PAD_LINK_FAILED (linkret)) {
    return GST_PAD_LINK_REFUSED;
  }

}

audioresample->channels = channels;
resample_set_n_channels (audioresample->resample, audioresample->channels);
if (pad == audioresample->srcpad) {
  audioresample->o_rate = rate;
  resample_set_output_rate (audioresample->resample, audioresample->o_rate);
  GST_DEBUG ("set o_rate to %d", rate);
} else {
  audioresample->i_rate = rate;
  resample_set_input_rate (audioresample->resample, audioresample->i_rate);
  GST_DEBUG ("set i_rate to %d", rate);
}

return GST_PAD_LINK_OK;
}

static void gst_audioresample_init (Audioresample * audioresample)
{
```

```
  ResampleState *r;

  audioresample->sinkpad =
      gst_pad_new_from_static_template (&gst_audioresample_sink_template,
      "sink");
  gst_element_add_pad (GST_ELEMENT (audioresample), audioresample->sinkpad);
  gst_pad_set_chain_function (audioresample->sinkpad, gst_audioresample_chain);
  gst_pad_set_link_function (audioresample->sinkpad, gst_audioresample_link);
  gst_pad_set_getcaps_function (audioresample->sinkpad,
      gst_audioresample_getcaps);
  gst_pad_set_fixate_function (audioresample->sinkpad,
      gst_audioresample_fixate);

  audioresample->srcpad =
      gst_pad_new_from_static_template (&gst_audioresample_src_template, "src");

  gst_element_add_pad (GST_ELEMENT (audioresample), audioresample->srcpad);
  gst_pad_set_link_function (audioresample->srcpad, gst_audioresample_link);
  gst_pad_set_getcaps_function (audioresample->srcpad,
      gst_audioresample_getcaps);
  gst_pad_set_fixate_function (audioresample->srcpad, gst_audioresample_fixate);

  r = resample_new ();
  audioresample->resample = r;

  resample_set_filter_length (r, 64);
  resample_set_format (r, RESAMPLE_FORMAT_S16);
}

static void gst_audioresample_dispose (GObject * object)
{
 Audioresample *audioresample = GST_AUDIORESAMPLE (object);

 if (audioresample->resample) {
   resample_free (audioresample->resample);
 }

 G_OBJECT_CLASS (parent_class)->dispose (object);
}

static void gst_audioresample_chain (GstPad * pad, GstData * _data)
{
 GstBuffer *buf = GST_BUFFER (_data);
 Audioresample *audioresample;
 ResampleState *r;
 guchar *data;
 gulong size;
 int outsize;
```

```
  GstBuffer *outbuf;

  g_return_if_fail (pad != NULL);
  g_return_if_fail (GST_IS_PAD (pad));
  g_return_if_fail (buf != NULL);

  audioresample = GST_AUDIORESAMPLE (gst_pad_get_parent (pad));

  if (!GST_IS_BUFFER (_data)) {
    gst_pad_push (audioresample->srcpad, _data);
    return;
  }

  if (audioresample->passthru) {
    gst_pad_push (audioresample->srcpad, GST_DATA (buf));
    return;
  }

  r = audioresample->resample;

  data = GST_BUFFER_DATA (buf);
  size = GST_BUFFER_SIZE (buf);

  GST_DEBUG ("got buffer of %ld bytes", size);

  resample_add_input_data (r, data, size, (ResampleCallback) gst_data_unref,
      buf);

  outsize = resample_get_output_size (r);
  /* FIXME this is audioresample being dumb.  dunno why */
  if (outsize == 0) {
    GST_ERROR ("overriding outbuf size");
    outsize = size;
  }
  outbuf = gst_buffer_new_and_alloc (outsize);

  outsize = resample_get_output_data (r, GST_BUFFER_DATA (outbuf), outsize);
  GST_BUFFER_SIZE (outbuf) = outsize;

  GST_BUFFER_TIMESTAMP (outbuf) =
      audioresample->offset * GST_SECOND / audioresample->o_rate;
  audioresample->offset += outsize / sizeof (gint16) / audioresample->channels;

  gst_pad_push (audioresample->srcpad, GST_DATA (outbuf));
}

static void
  gst_audioresample_set_property (GObject * object, guint prop_id,
```

```
    const GValue * value, GParamSpec * pspec)
{
 Audioresample *audioresample;

   g_return_if_fail (GST_IS_AUDIORESAMPLE (object));
   audioresample = GST_AUDIORESAMPLE (object);

 switch (prop_id) {
   case ARG_FILTERLEN:
     audioresample->filter_length = g_value_get_int (value);
     GST_DEBUG_OBJECT (GST_ELEMENT (audioresample), "new filter length %d\n",
       audioresample->filter_length);
     resample_set_filter_length (audioresample->resample,
       audioresample->filter_length);
     break;
     default:G_OBJECT_WARN_INVALID_PROPERTY_ID (object, prop_id, pspec);
     break;
 }
}

static void
  gst_audioresample_get_property (GObject * object, guint prop_id,
  GValue * value, GParamSpec * pspec)
{
 Audioresample *audioresample;

 g_return_if_fail (GST_IS_AUDIORESAMPLE (object));
 audioresample = GST_AUDIORESAMPLE (object);

 switch (prop_id) {
   case ARG_FILTERLEN:
     g_value_set_int (value, audioresample->filter_length);
     break;
   default:
     G_OBJECT_WARN_INVALID_PROPERTY_ID (object, prop_id, pspec);
     break;
 }
}


static gboolean plugin_init (GstPlugin * plugin)
{
 resample_init ();

 if (!gst_element_register (plugin, "audioresample", GST_RANK_PRIMARY,
     GST_TYPE_AUDIORESAMPLE)) {
  return FALSE;
 }
```

```
 return TRUE;
}

GST_PLUGIN_DEFINE (GST_VERSION_MAJOR,
  GST_VERSION_MINOR,
  "audioresample",
  "Resamples audio", plugin_init, VERSION, "LGPL", GST_PACKAGE_NAME,
  GST_PACKAGE_ORIGIN)
/* GStreamer
* Copyright (C) <1999> Erik Walthinsen <omega@cse.ogi.edu>
*
* 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., 59 Temple Place - Suite 330,
* Boston, MA 02111-1307, USA.
*/


#ifndef __AUDIORESAMPLE_H__
#define __AUDIORESAMPLE_H__


#include <gst/gst.h>

#include <audioresample/resample.h>


G_BEGIN_DECLS


#define GST_TYPE_AUDIORESAMPLE \
 (audioresample_get_type())
#define GST_AUDIORESAMPLE(obj) \
 (G_TYPE_CHECK_INSTANCE_CAST((obj),GST_TYPE_AUDIORESAMPLE,Audioresample))
#define GST_AUDIORESAMPLE_CLASS(klass) \
 (G_TYPE_CHECK_CLASS_CAST((klass),GST_TYPE_AUDIORESAMPLE,AudioresampleClass))
#define GST_IS_AUDIORESAMPLE(obj) \
```

```
 (G_TYPE_CHECK_INSTANCE_TYPE((obj),GST_TYPE_AUDIORESAMPLE))
#define GST_IS_AUDIORESAMPLE_CLASS(klass) \
 (G_TYPE_CHECK_CLASS_TYPE((klass),GST_TYPE_AUDIORESAMPLE))

typedef struct _Audioresample Audioresample;
typedef struct _AudioresampleClass AudioresampleClass;

struct _Audioresample {
 GstElement element;

 GstPad *sinkpad,*srcpad;

 gboolean passthru;

 gint64 offset;
 int channels;

 int i_rate;
 int o_rate;
 int filter_length;

 ResampleState * resample;
};

struct _AudioresampleClass {
 GstElementClass parent_class;
};

GType gst_audioresample_get_type(void);


G_END_DECLS


#endif /* __AUDIORESAMPLE_H__ */
/* GStreamer
* Copyright (C) <1999> Erik Walthinsen <omega@cse.ogi.edu>
* Copyright (C) <2003> David Schleef <ds@schleef.org>
*
* 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., 59 Temple Place - Suite 330,
* Boston, MA 02111-1307, USA.
*/

/*
* This file was (probably) generated from
* gstvideotemplate.c,v 1.18 2005/11/14 02:13:34 thomasvs Exp
* and
* $Id: make_filter,v 1.8 2004/04/19 22:51:57 ds Exp $
*/

#ifdef HAVE_CONFIG_H
#include "config.h"
#endif

#include <gst/gst.h>
#include <gst/base/gstbasetransform.h>
#include <gst/video/video.h>
#include <string.h>
#include <cog/cog.h>
#include <math.h>
#include <cog/cogvirtframe.h>

#define GST_TYPE_COGDOWNSAMPLE \
  (gst_cogdownsample_get_type())
#define GST_COGDOWNSAMPLE(obj) \
  (G_TYPE_CHECK_INSTANCE_CAST((obj),GST_TYPE_COGDOWNSAMPLE,GstCogdownsample))
#define GST_COGDOWNSAMPLE_CLASS(klass) \
  (G_TYPE_CHECK_CLASS_CAST((klass),GST_TYPE_COGDOWNSAMPLE,GstCogdownsampleClass))
#define GST_IS_COGDOWNSAMPLE(obj) \
  (G_TYPE_CHECK_INSTANCE_TYPE((obj),GST_TYPE_COGDOWNSAMPLE))
#define GST_IS_COGDOWNSAMPLE_CLASS(obj) \
  (G_TYPE_CHECK_CLASS_TYPE((klass),GST_TYPE_COGDOWNSAMPLE))

typedef struct _GstCogdownsample GstCogdownsample;
typedef struct _GstCogdownsampleClass GstCogdownsampleClass;

struct _GstCogdownsample
{
  GstBaseTransform base_transform;

};

struct _GstCogdownsampleClass
{
```

```c
  GstBaseTransformClass parent_class;

};

GType gst_cogdownsample_get_type (void);

enum
{
 ARG_0
};

static void gst_cogdownsample_base_init (gpointer g_class);
static void gst_cogdownsample_class_init (gpointer g_class,
   gpointer class_data);
static void gst_cogdownsample_init (GTypeInstance * instance, gpointer g_class);

static void gst_cogdownsample_set_property (GObject * object, guint prop_id,
   const GValue * value, GParamSpec * pspec);
static void gst_cogdownsample_get_property (GObject * object, guint prop_id,
   GValue * value, GParamSpec * pspec);

static GstCaps *gst_cogdownsample_transform_caps (GstBaseTransform *
   base_transform, GstPadDirection direction, GstCaps * caps);
static GstFlowReturn gst_cogdownsample_transform (GstBaseTransform *
   base_transform, GstBuffer * inbuf, GstBuffer * outbuf);
static gboolean gst_cogdownsample_get_unit_size (GstBaseTransform *
   base_transform, GstCaps * caps, guint * size);

static GstStaticPadTemplate gst_cogdownsample_sink_template =
GST_STATIC_PAD_TEMPLATE ("sink",
  GST_PAD_SINK,
  GST_PAD_ALWAYS,
  GST_STATIC_CAPS (GST_VIDEO_CAPS_YUV ("{ I420, YV12, YUY2, UYVY, AYUV }"))
  );

static GstStaticPadTemplate gst_cogdownsample_src_template =
GST_STATIC_PAD_TEMPLATE ("src",
  GST_PAD_SRC,
  GST_PAD_ALWAYS,
  GST_STATIC_CAPS (GST_VIDEO_CAPS_YUV ("{ I420, YV12, YUY2, UYVY, AYUV }"))
  );

GType
gst_cogdownsample_get_type (void)
{
 static GType compress_type = 0;

 if (!compress_type) {
```

```c
  static const GTypeInfo compress_info = {
    sizeof (GstCogdownsampleClass),
    gst_cogdownsample_base_init,
    NULL,
    gst_cogdownsample_class_init,
    NULL,
    NULL,
    sizeof (GstCogdownsample),
    0,
    gst_cogdownsample_init,
  };

  compress_type = g_type_register_static (GST_TYPE_BASE_TRANSFORM,
      "GstCogdownsample", &compress_info, 0);
}
return compress_type;
}


static void
gst_cogdownsample_base_init (gpointer g_class)
{

GstElementClass *element_class = GST_ELEMENT_CLASS (g_class);

gst_element_class_add_pad_template (element_class,
    gst_static_pad_template_get (&gst_cogdownsample_src_template));
gst_element_class_add_pad_template (element_class,
    gst_static_pad_template_get (&gst_cogdownsample_sink_template));

gst_element_class_set_details_simple (element_class,
    "Scale down video by factor of 2", "Filter/Effect/Video",
    "Scales down video by a factor of 2", "David Schleef <ds@schleef.org>");
}

static void
gst_cogdownsample_class_init (gpointer g_class, gpointer class_data)
{
GObjectClass *gobject_class;
GstBaseTransformClass *base_transform_class;

gobject_class = G_OBJECT_CLASS (g_class);
base_transform_class = GST_BASE_TRANSFORM_CLASS (g_class);

gobject_class->set_property = gst_cogdownsample_set_property;
gobject_class->get_property = gst_cogdownsample_get_property;

base_transform_class->transform = gst_cogdownsample_transform;
```

```
base_transform_class->transform_caps = gst_cogdownsample_transform_caps;
base_transform_class->get_unit_size = gst_cogdownsample_get_unit_size;
}

static void
gst_cogdownsample_init (GTypeInstance * instance, gpointer g_class)
{

GST_DEBUG ("gst_cogdownsample_init");
}

static void
gst_cogdownsample_set_property (GObject * object, guint prop_id,
   const GValue * value, GParamSpec * pspec)
{
GstCogdownsample *src;

 g_return_if_fail (GST_IS_COGDOWNSAMPLE (object));
 src = GST_COGDOWNSAMPLE (object);

 GST_DEBUG ("gst_cogdownsample_set_property");
 switch (prop_id) {
  default:
    break;
 }
}

static void
gst_cogdownsample_get_property (GObject * object, guint prop_id, GValue * value,
   GParamSpec * pspec)
{
GstCogdownsample *src;

 g_return_if_fail (GST_IS_COGDOWNSAMPLE (object));
 src = GST_COGDOWNSAMPLE (object);

 switch (prop_id) {
  default:
    G_OBJECT_WARN_INVALID_PROPERTY_ID (object, prop_id, pspec);
    break;
 }
}

static void
transform_value (GValue * dest, const GValue * src, GstPadDirection dir)
{
 g_value_init (dest, G_VALUE_TYPE (src));
```

```c
  if (G_VALUE_HOLDS_INT (src)) {
    int x;

    x = g_value_get_int (src);
    if (dir == GST_PAD_SINK) {
      g_value_set_int (dest, x / 2);
    } else {
      g_value_set_int (dest, x * 2);
    }
  } else if (GST_VALUE_HOLDS_INT_RANGE (src)) {
    int min, max;

    min = gst_value_get_int_range_min (src);
    max = gst_value_get_int_range_max (src);

    if (dir == GST_PAD_SINK) {
      min = (min + 1) / 2;
      if (max == G_MAXINT) {
        max = G_MAXINT / 2;
      } else {
        max = (max + 1) / 2;
      }
    } else {
      if (max > G_MAXINT / 2) {
        max = G_MAXINT;
      } else {
        max = max * 2;
      }
      if (min > G_MAXINT / 2) {
        min = G_MAXINT;
      } else {
        min = min * 2;
      }
    }
    gst_value_set_int_range (dest, min, max);
  } else {
    /* FIXME */
    g_warning ("case not handled");
    g_value_set_int (dest, 100);
  }
}

static GstCaps *
gst_cogdownsample_transform_caps (GstBaseTransform * base_transform,
    GstPadDirection direction, GstCaps * caps)
{
  int i;
  GstStructure *structure;
```

```
  GValue new_value = { 0 };
  const GValue *value;

  caps = gst_caps_copy (caps);

  for (i = 0; i < gst_caps_get_size (caps); i++) {
    structure = gst_caps_get_structure (caps, i);

    value = gst_structure_get_value (structure, "width");
    transform_value (&new_value, value, direction);
    gst_structure_set_value (structure, "width", &new_value);
    g_value_unset (&new_value);

    value = gst_structure_get_value (structure, "height");
    transform_value (&new_value, value, direction);
    gst_structure_set_value (structure, "height", &new_value);
    g_value_unset (&new_value);
  }

  return caps;
}

static gboolean
gst_cogdownsample_get_unit_size (GstBaseTransform * base_transform,
    GstCaps * caps, guint * size)
{
  int width, height;
  uint32_t format;

  gst_structure_get_fourcc (gst_caps_get_structure (caps, 0),
      "format", &format);
  gst_structure_get_int (gst_caps_get_structure (caps, 0), "width", &width);
  gst_structure_get_int (gst_caps_get_structure (caps, 0), "height", &height);

  switch (format) {
    case GST_MAKE_FOURCC ('I', '4', '2', '0'):
    case GST_MAKE_FOURCC ('Y', 'V', '1', '2'):
      *size = width * height * 3 / 2;
      break;
    case GST_MAKE_FOURCC ('Y', 'U', 'Y', '2'):
    case GST_MAKE_FOURCC ('U', 'Y', 'V', 'Y'):
      *size = width * height * 2;
      break;
    case GST_MAKE_FOURCC ('A', 'Y', 'U', 'V'):
      *size = width * height * 4;
      break;
    default:
      g_assert_not_reached ();
```

```
    }

  return TRUE;
}

static GstFlowReturn
gst_cogdownsample_transform (GstBaseTransform * base_transform,
    GstBuffer * inbuf, GstBuffer * outbuf)
{
  GstCogdownsample *compress;
  CogFrame *outframe;
  int width, height;
  uint32_t format;
  CogFrame *frame;

  g_return_val_if_fail (GST_IS_COGDOWNSAMPLE (base_transform), GST_FLOW_ERROR);
  compress = GST_COGDOWNSAMPLE (base_transform);

  gst_structure_get_fourcc (gst_caps_get_structure (inbuf->caps, 0),
      "format", &format);
  gst_structure_get_int (gst_caps_get_structure (inbuf->caps, 0),
      "width", &width);
  gst_structure_get_int (gst_caps_get_structure (inbuf->caps, 0),
      "height", &height);

  switch (format) {
    case GST_MAKE_FOURCC ('I', '4', '2', '0'):
      frame = cog_frame_new_from_data_I420 (GST_BUFFER_DATA (inbuf),
          width, height);
      outframe = cog_frame_new_from_data_I420 (GST_BUFFER_DATA (outbuf),
          width / 2, height / 2);
      break;
    case GST_MAKE_FOURCC ('Y', 'V', '1', '2'):
      frame = cog_frame_new_from_data_YV12 (GST_BUFFER_DATA (inbuf),
          width, height);
      outframe = cog_frame_new_from_data_YV12 (GST_BUFFER_DATA (outbuf),
          width / 2, height / 2);
      break;
    case GST_MAKE_FOURCC ('Y', 'U', 'Y', '2'):
      frame = cog_frame_new_from_data_YUY2 (GST_BUFFER_DATA (inbuf),
          width, height);
      outframe = cog_frame_new_from_data_YUY2 (GST_BUFFER_DATA (outbuf),
          width / 2, height / 2);
      break;
    case GST_MAKE_FOURCC ('U', 'Y', 'V', 'Y'):
      frame = cog_frame_new_from_data_UYVY (GST_BUFFER_DATA (inbuf),
          width, height);
      outframe = cog_frame_new_from_data_UYVY (GST_BUFFER_DATA (outbuf),
```

```
      width / 2, height / 2);
    break;
  case GST_MAKE_FOURCC ('A', 'Y', 'U', 'V'):
    frame = cog_frame_new_from_data_AYUV (GST_BUFFER_DATA (inbuf),
        width, height);
    outframe = cog_frame_new_from_data_AYUV (GST_BUFFER_DATA (outbuf),
        width / 2, height / 2);
    break;
  default:
    g_assert_not_reached ();
    return GST_FLOW_ERROR;
}

frame = cog_virt_frame_new_unpack (frame);
frame = cog_virt_frame_new_horiz_downsample (frame, 3);
frame = cog_virt_frame_new_vert_downsample (frame, 2);

switch (format) {
  case GST_MAKE_FOURCC ('Y', 'U', 'Y', '2'):
    frame = cog_virt_frame_new_pack_YUY2 (frame);
    break;
  case GST_MAKE_FOURCC ('U', 'Y', 'V', 'Y'):
    frame = cog_virt_frame_new_pack_UYVY (frame);
    break;
  case GST_MAKE_FOURCC ('A', 'Y', 'U', 'V'):
    frame = cog_virt_frame_new_pack_AYUV (frame);
    break;
  default:
    break;
}

cog_virt_frame_render (frame, outframe);
cog_frame_unref (frame);
cog_frame_unref (outframe);

return GST_FLOW_OK;
}
/* GStreamer mplex (mjpegtools) wrapper
* (c) 2003 Ronald Bultje <rbultje@ronald.bitfreak.net>
* (c) 2008 Mark Nauwelaerts <mnauw@users.sourceforge.net>
*
* gstmplex.cc: gstreamer mplex wrapper
*
* 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., 59 Temple Place - Suite 330,
 * Boston, MA 02111-1307, USA.
 */

/**
 * SECTION:element-mplex
 * @see_also: mpeg2enc
 *
 * This element is an audio/video multiplexer for MPEG-1/2 video streams
 * and (un)compressed audio streams such as AC3, MPEG layer I/II/III.
 * It is based on the <ulink url="http://mjpeg.sourceforge.net/">mjpegtools</ulink> library.
 * Documentation on creating MPEG videos in general can be found in the
 * <ulink url="https://sourceforge.net/docman/display_doc.php?docid=3456&group_id=5776">MJPEG
Howto</ulink>
 * and the man-page of the mplex tool documents the properties of this element,
 * which are shared with the mplex tool.
 *
 * <refsect2>
 * <title>Example pipeline</title>
 * |[
 * gst-launch -v videotestsrc num-buffers=1000 ! mpeg2enc ! mplex ! filesink location=videotestsrc.mpg
 * ]| This example pipeline will encode a test video source to an
 * MPEG1 elementary stream and multiplexes this to an MPEG system stream.
 * <para>
 * If several streams are being multiplexed, there should (as usual) be
 * a queue in each stream, and due to mplex' buffering the capacities of these
 * may have to be set to a few times the default settings to prevent the
 * pipeline stalling.
 * </para>
 * </refsect2>
 */

#ifdef HAVE_CONFIG_H
#include "config.h"
#endif

#include <string.h>

#include "gstmplex.hh"
#include "gstmplexoutputstream.hh"
#include "gstmplexibitstream.hh"
```

```c
#include "gstmplexjob.hh"

GST_DEBUG_CATEGORY (mplex_debug);

static GstStaticPadTemplate src_templ = GST_STATIC_PAD_TEMPLATE ("src",
  GST_PAD_SRC,
  GST_PAD_ALWAYS,
  GST_STATIC_CAPS ("video/mpeg, systemstream = (boolean) true ")
  );

static GstStaticPadTemplate video_sink_templ =
GST_STATIC_PAD_TEMPLATE ("video_%d",
  GST_PAD_SINK,
  GST_PAD_REQUEST,
  GST_STATIC_CAPS ("video/mpeg, "
    "mpegversion = (int) { 1, 2 }, "
    "systemstream = (boolean) false, "
    "width = (int) [ 16, 4096 ], "
    "height = (int) [ 16, 4096 ], framerate = (fraction) [ 0, MAX ]")
  );

#define COMMON_AUDIO_CAPS \
 "channels = (int) [ 1, 8 ], " \
 "rate = (int) [ 8000, 96000 ]"

static GstStaticPadTemplate audio_sink_templ =
  GST_STATIC_PAD_TEMPLATE ("audio_%d",
  GST_PAD_SINK,
  GST_PAD_REQUEST,
  GST_STATIC_CAPS ("audio/mpeg, "
    "mpegversion = (int) 1, "
    "layer = (int) [ 1, 3 ], "
    COMMON_AUDIO_CAPS "; "
    "audio/x-ac3, "
    COMMON_AUDIO_CAPS "; "
    "audio/x-dts; "
    "audio/x-raw-int, "
    "endianness = (int) BIG_ENDIAN, "
    "signed = (boolean) TRUE, "
    "width = (int) { 16, 20, 24 }, "
    "depth = (int) { 16, 20, 24 }, "
    "rate = (int) { 48000, 96000 }, " "channels = (int) [ 1, 6 ]")
  );

/* FIXME: subtitles */

static void gst_mplex_finalize (GObject * object);
static void gst_mplex_reset (GstMplex * mplex);
```

```c
static void gst_mplex_loop (GstMplex * mplex);
static GstPad *gst_mplex_request_new_pad (GstElement * element,
    GstPadTemplate * templ, const gchar * name);
static void gst_mplex_release_pad (GstElement * element, GstPad * pad);
static gboolean gst_mplex_src_activate_push (GstPad * pad, gboolean active);
static GstStateChangeReturn gst_mplex_change_state (GstElement * element,
    GstStateChange transition);

static void gst_mplex_get_property (GObject * object,
    guint prop_id, GValue * value, GParamSpec * pspec);
static void gst_mplex_set_property (GObject * object,
    guint prop_id, const GValue * value, GParamSpec * pspec);

GST_BOILERPLATE (GstMplex, gst_mplex, GstElement, GST_TYPE_ELEMENT);

static void
gst_mplex_base_init (gpointer klass)
{
 GstElementClass *element_class = GST_ELEMENT_CLASS (klass);

 gst_element_class_set_details_simple (element_class,
     "mplex video multiplexer", "Codec/Muxer",
     "High-quality MPEG/DVD/SVCD/VCD video/audio multiplexer",
     "Andrew Stevens <andrew.stevens@nexgo.de>\n"
     "Ronald Bultje <rbultje@ronald.bitfreak.net>\n"
     "Mark Nauwelaerts <mnauw@users.sourceforge.net>");

 gst_element_class_add_pad_template (element_class,
     gst_static_pad_template_get (&src_templ));
 gst_element_class_add_pad_template (element_class,
     gst_static_pad_template_get (&video_sink_templ));
 gst_element_class_add_pad_template (element_class,
     gst_static_pad_template_get (&audio_sink_templ));
}

static void
gst_mplex_class_init (GstMplexClass * klass)
{
 GObjectClass *object_class = G_OBJECT_CLASS (klass);
 GstElementClass *element_class = GST_ELEMENT_CLASS (klass);

 GST_DEBUG_CATEGORY_INIT (mplex_debug, "mplex", 0, "MPEG video/audio muxer");

 object_class->set_property = gst_mplex_set_property;
 object_class->get_property = gst_mplex_get_property;

 /* register properties */
 GstMplexJob::initProperties (object_class);
```

```
  object_class->finalize = GST_DEBUG_FUNCPTR (gst_mplex_finalize);

  element_class->change_state = GST_DEBUG_FUNCPTR (gst_mplex_change_state);
  element_class->request_new_pad =
      GST_DEBUG_FUNCPTR (gst_mplex_request_new_pad);
  element_class->release_pad = GST_DEBUG_FUNCPTR (gst_mplex_release_pad);
}

static void
gst_mplex_finalize (GObject * object)
{
 GstMplex *mplex = GST_MPLEX (object);
 GSList *walk;

 /* release all pads */
 walk = mplex->pads;
 while (walk) {
  GstMplexPad *mpad = (GstMplexPad *) walk->data;

  gst_object_unref (mpad->pad);
  mpad->pad = NULL;
  walk = walk->next;
 }

 /* clean up what's left of them */
 gst_mplex_reset (mplex);

 /* ... and of the rest */
 delete mplex->job;

 g_mutex_free (mplex->tlock);

 G_OBJECT_CLASS (parent_class)->finalize (object);
}

static void
gst_mplex_init (GstMplex * mplex, GstMplexClass * g_class)
{
 GstElement *element = GST_ELEMENT (mplex);
 GstElementClass *element_class = GST_ELEMENT_CLASS (g_class);

 mplex->srcpad =
     gst_pad_new_from_template (gst_element_class_get_pad_template
     (element_class, "src"), "src");
 gst_element_add_pad (element, mplex->srcpad);
 gst_pad_use_fixed_caps (mplex->srcpad);
 gst_pad_set_activatepush_function (mplex->srcpad,
```

```
  GST_DEBUG_FUNCPTR (gst_mplex_src_activate_push));

mplex->job = new GstMplexJob ();
mplex->num_apads = 0;
mplex->num_vpads = 0;

mplex->tlock = g_mutex_new ();

gst_mplex_reset (mplex);
}

static void
gst_mplex_reset (GstMplex * mplex)
{
 GSList *walk;
 GSList *nlist = NULL;

 mplex->eos = FALSE;
 mplex->srcresult = GST_FLOW_CUSTOM_SUCCESS;

 /* reset existing streams */
 walk = mplex->pads;
 while (walk != NULL) {
  GstMplexPad *mpad;

  mpad = (GstMplexPad *) walk->data;

  mpad->needed = 0;
  mpad->eos = FALSE;
  gst_adapter_clear (mpad->adapter);
  if (mpad->bs) {
   delete mpad->bs;

   mpad->bs = NULL;
  }

  if (!mpad->pad) {
   g_cond_free (mpad->cond);
   g_object_unref (mpad->adapter);
   g_free (mpad);
  } else
   nlist = g_slist_append (nlist, mpad);

  walk = walk->next;
 }

 g_slist_free (mplex->pads);
 mplex->pads = nlist;
```

```
/* clear mplex stuff */
/* clean up stream settings */
while (!mplex->job->streams.empty ()) {
  delete mplex->job->streams.back ();

  mplex->job->streams.pop_back ();
}
while (!mplex->job->video_param.empty ()) {
  delete mplex->job->video_param.back ();

  mplex->job->video_param.pop_back ();
}
while (!mplex->job->lpcm_param.empty ()) {
  delete mplex->job->lpcm_param.back ();

  mplex->job->lpcm_param.pop_back ();
}
mplex->job->audio_tracks = 0;
mplex->job->video_tracks = 0;
mplex->job->lpcm_tracks = 0;
}

static gboolean
gst_mplex_setcaps (GstPad * pad, GstCaps * caps)
{
 GstMplex *mplex;
 const gchar *mime;
 GstStructure *structure;
 StreamKind type;
 JobStream *jobstream;
 GstMplexIBitStream *inputstream;
 GstMplexPad *mpad;
 GstCaps *othercaps;
 gboolean ret = TRUE;

 mplex = GST_MPLEX (GST_PAD_PARENT (pad));

 /* does not go well to negotiate when started */
 if (mplex->srcresult != GST_FLOW_CUSTOM_SUCCESS)
   goto refuse_renegotiation;

 /* since muxer does not really check much ... */
 othercaps = gst_caps_intersect (caps, gst_pad_get_pad_template_caps (pad));
 if (othercaps)
   gst_caps_unref (othercaps);
 else
   goto refuse_caps;
```

```
/* set the fixed template caps on the srcpad, should accept without objection */
othercaps = gst_caps_copy (gst_pad_get_pad_template_caps (mplex->srcpad));
ret = gst_pad_set_caps (mplex->srcpad, othercaps);
gst_caps_unref (othercaps);
if (!ret)
  goto refuse_caps;

structure = gst_caps_get_structure (caps, 0);
mime = gst_structure_get_name (structure);

if (!strcmp (mime, "video/mpeg")) {   /* video */
  VideoParams *params;

  type = MPEG_VIDEO;
  if (mplex->job->bufsize)
    params = VideoParams::Checked (mplex->job->bufsize);
  else
    params = VideoParams::Default (mplex->job->mux_format);
  /* set standard values if forced by the selected profile */
  if (params->Force (mplex->job->mux_format))
    GST_WARNING_OBJECT (mplex,
        "overriding non-standard option due to selected profile");

  mplex->job->video_param.push_back (params);
  mplex->job->video_tracks++;
} else {                  /* audio */
  if (!strcmp (mime, "audio/mpeg")) {
    type = MPEG_AUDIO;
  } else if (!strcmp (mime, "audio/x-ac3")) {
    type = AC3_AUDIO;
  } else if (!strcmp (mime, "audio/x-dts")) {
    type = DTS_AUDIO;
  } else if (!strcmp (mime, "audio/x-raw-int")) {
    LpcmParams *params;
    gint bits, chans, rate;
    gboolean result = TRUE;

    type = LPCM_AUDIO;

    /* set LPCM params */
    result &= gst_structure_get_int (structure, "depth", &bits);
    result &= gst_structure_get_int (structure, "rate", &rate);
    result &= gst_structure_get_int (structure, "channels", &chans);
    if (!result)
      goto refuse_caps;

    params = LpcmParams::Checked (rate, chans, bits);
```

```
        mplex->job->lpcm_param.push_back (params);
        mplex->job->lpcm_tracks++;
      } else
        goto refuse_caps;

      mplex->job->audio_tracks++;
    }

    mpad = (GstMplexPad *) gst_pad_get_element_private (pad);
    g_return_val_if_fail (mpad, FALSE);
    inputstream = new GstMplexIBitStream (mpad);
    mpad->bs = inputstream;
    jobstream = new JobStream (inputstream, type);
    mplex->job->streams.push_back (jobstream);

    return TRUE;

  refuse_caps:
    {
      GST_WARNING_OBJECT (mplex, "refused caps %" GST_PTR_FORMAT, caps);

      /* undo if we were a bit too fast/confident */
      if (GST_PAD_CAPS (mplex->srcpad))
        gst_pad_set_caps (mplex->srcpad, NULL);

      return FALSE;
    }
  refuse_renegotiation:
    {
      GST_WARNING_OBJECT (mplex, "already started; "
          "refused (re)negotiation (to %" GST_PTR_FORMAT ")", caps);

      return FALSE;
    }
  }

  static void
  gst_mplex_loop (GstMplex * mplex)
  {
    GstMplexOutputStream *out = NULL;
    Multiplexor *mux = NULL;
    GSList *walk;

    /* do not try to resume muxing after it finished
     * this can be relevant mainly/only in case of forced state change */
    if (mplex->eos)
      goto eos;
```

```
  /* inform downstream about what's coming */
  gst_pad_push_event (mplex->srcpad, gst_event_new_new_segment (FALSE, 1.0,
      GST_FORMAT_BYTES, 0, -1, 0));

  /* hm (!) each inputstream really needs an initial read
   * so that all is internally in the proper state */
  walk = mplex->pads;
  while (walk != NULL) {
    GstMplexPad *mpad;

    mpad = (GstMplexPad *) walk->data;
    mpad->bs->ReadBuffer ();

    walk = walk->next;
  }

  /* create new multiplexer with inputs/output */
  out = new GstMplexOutputStream (mplex, mplex->srcpad);
#if GST_MJPEGTOOLS_API >= 10900
  mux = new Multiplexor (*mplex->job, *out, NULL);
#else
  mux = new Multiplexor (*mplex->job, *out);
#endif

  if (mux) {
    mux->Multiplex ();
    delete mux;
    delete out;

    /* if not well and truly eos, something strange happened  */
    if (!mplex->eos) {
      GST_ERROR_OBJECT (mplex, "muxing task ended without being eos");
      /* notify there is no point in collecting any more */
      GST_MPLEX_MUTEX_LOCK (mplex);
      mplex->srcresult = GST_FLOW_ERROR;
      GST_MPLEX_SIGNAL_ALL (mplex);
      GST_MPLEX_MUTEX_UNLOCK (mplex);
    } else
      goto eos;
  } else {
    GST_WARNING_OBJECT (mplex, "failed to create Multiplexor");
  }

  /* fall-through */
done:
  {
    /* no need to run wildly, stopped elsewhere, e.g. state change */
```

```
      GST_DEBUG_OBJECT (mplex, "pausing muxing task");
      gst_pad_pause_task (mplex->srcpad);

      return;
    }
  eos:
    {
      GST_DEBUG_OBJECT (mplex, "encoding task reached eos");
      goto done;
    }
  }

static gboolean
gst_mplex_sink_event (GstPad * sinkpad, GstEvent * event)
{
  GstMplex *mplex;
  GstMplexPad *mpad;
  gboolean result = TRUE;

  mplex = (GstMplex *) (GST_PAD_PARENT (sinkpad));
  mpad = (GstMplexPad *) gst_pad_get_element_private (sinkpad);
  g_return_val_if_fail (mpad, FALSE);

  switch (GST_EVENT_TYPE (event)) {
    case GST_EVENT_FLUSH_START:
      /* forward event */
      gst_pad_event_default (sinkpad, event);

      /* now unblock the chain function */
      GST_MPLEX_MUTEX_LOCK (mplex);
      mplex->srcresult = GST_FLOW_WRONG_STATE;
      GST_MPLEX_SIGNAL (mplex, mpad);
      GST_MPLEX_MUTEX_UNLOCK (mplex);
      /* no way to pause/restart loop task */
      goto done;
    case GST_EVENT_FLUSH_STOP:
      /* forward event */
      gst_pad_event_default (sinkpad, event);

      /* clear state and resume */
      GST_MPLEX_MUTEX_LOCK (mplex);
      gst_adapter_clear (mpad->adapter);
      mplex->srcresult = GST_FLOW_OK;
      GST_MPLEX_MUTEX_UNLOCK (mplex);
      goto done;
    case GST_EVENT_NEWSEGMENT:
      /* eat segments; we make our own (byte)stream */
      gst_event_unref (event);
```

```
      goto done;
    case GST_EVENT_EOS:
      /* inform this pad that it can stop now */
      GST_MPLEX_MUTEX_LOCK (mplex);
      mpad->eos = TRUE;
      GST_MPLEX_SIGNAL (mplex, mpad);
      GST_MPLEX_MUTEX_UNLOCK (mplex);

      /* eat this event for now, task will send eos when finished */
      gst_event_unref (event);
      goto done;
    default:
      /* for a serialized event, wait until earlier data is gone,
       * though this is no guarantee as to when task is done with it.
       * Only wait if loop has been started already */
      if (GST_EVENT_IS_SERIALIZED (event)) {
        GST_MPLEX_MUTEX_LOCK (mplex);
        while (mplex->srcresult == GST_FLOW_OK && !mpad->needed)
          GST_MPLEX_WAIT (mplex, mpad);
        GST_MPLEX_MUTEX_UNLOCK (mplex);
      }
      break;
  }

  result = gst_pad_event_default (sinkpad, event);

done:
  return result;
}

/* starts task if conditions are right for it
 * must be called with mutex_lock held */
static void
gst_mplex_start_task (GstMplex * mplex)
{
  /* start task to create multiplexor and start muxing */
  if (G_UNLIKELY (mplex->srcresult == GST_FLOW_CUSTOM_SUCCESS)
      && mplex->job->video_tracks == mplex->num_vpads
      && mplex->job->audio_tracks == mplex->num_apads) {
    gst_pad_start_task (mplex->srcpad, (GstTaskFunction) gst_mplex_loop, mplex);
    mplex->srcresult = GST_FLOW_OK;
  }
}

static GstFlowReturn
gst_mplex_chain (GstPad * sinkpad, GstBuffer * buffer)
{
  GstMplex *mplex;
```

```c
  GstMplexPad *mpad;

  mplex = (GstMplex *) (GST_PAD_PARENT (sinkpad));
  mpad = (GstMplexPad *) gst_pad_get_element_private (sinkpad);
  g_return_val_if_fail (mpad, GST_FLOW_ERROR);

  /* check if pad were properly negotiated and set up */
  if (G_UNLIKELY (!mpad->bs)) {
    GST_ELEMENT_ERROR (mplex, CORE, NEGOTIATION, (NULL),
        ("input pad has not been set up prior to chain function"));
    return GST_FLOW_NOT_NEGOTIATED;
  }

  GST_MPLEX_MUTEX_LOCK (mplex);

  gst_mplex_start_task (mplex);

  if (G_UNLIKELY (mpad->eos))
    goto eos;

  if (G_UNLIKELY (mplex->srcresult != GST_FLOW_OK))
    goto ignore;

  gst_adapter_push (mpad->adapter, buffer);
  buffer = NULL;
  while (gst_adapter_available (mpad->adapter) >= mpad->needed) {
    GST_MPLEX_SIGNAL (mplex, mpad);
    GST_MPLEX_WAIT (mplex, mpad);
    /* may have become flushing or in error */
    if (G_UNLIKELY (mplex->srcresult != GST_FLOW_OK))
      goto ignore;
    /* or been removed */
    if (G_UNLIKELY (mpad->eos))
      goto eos;
  }

  GST_MPLEX_MUTEX_UNLOCK (mplex);

  return GST_FLOW_OK;

  /* special cases */
eos:
  {
    GST_DEBUG_OBJECT (mplex, "ignoring buffer at end-of-stream");
    GST_MPLEX_MUTEX_UNLOCK (mplex);

    gst_buffer_unref (buffer);
    return GST_FLOW_UNEXPECTED;
```

```
  }
ignore:
 {
  GstFlowReturn ret = mplex->srcresult;

  GST_DEBUG_OBJECT (mplex, "ignoring buffer because src task encountered %s",
     gst_flow_get_name (ret));
  GST_MPLEX_MUTEX_UNLOCK (mplex);

  if (buffer)
    gst_buffer_unref (buffer);
  return ret;
 }
}

static GstPad *
gst_mplex_request_new_pad (GstElement * element,
  GstPadTemplate * templ, const gchar * name)
{
 GstElementClass *klass = GST_ELEMENT_GET_CLASS (element);
 GstMplex *mplex = GST_MPLEX (element);
 gchar *padname;
 GstPad *newpad;
 GstMplexPad *mpad;

 if (templ == gst_element_class_get_pad_template (klass, "audio_%d")) {
   GST_DEBUG_OBJECT (mplex, "request pad audio %d", mplex->num_apads);
   padname = g_strdup_printf ("audio_%d", mplex->num_apads++);
 } else if (templ == gst_element_class_get_pad_template (klass, "video_%d")) {
   GST_DEBUG_OBJECT (mplex, "request pad video %d", mplex->num_vpads);
   padname = g_strdup_printf ("video_%d", mplex->num_vpads++);
 } else {
   GST_WARNING_OBJECT (mplex, "This is not our template!");
   return NULL;
 }

 newpad = gst_pad_new_from_template (templ, padname);
 g_free (padname);

 mpad = g_new0 (GstMplexPad, 1);
 mpad->adapter = gst_adapter_new ();
 mpad->cond = g_cond_new ();
 gst_object_ref (newpad);
 mpad->pad = newpad;

 gst_pad_set_setcaps_function (newpad, GST_DEBUG_FUNCPTR (gst_mplex_setcaps));
 gst_pad_set_chain_function (newpad, GST_DEBUG_FUNCPTR (gst_mplex_chain));
 gst_pad_set_event_function (newpad, GST_DEBUG_FUNCPTR (gst_mplex_sink_event));
```

```
  gst_pad_set_element_private (newpad, mpad);
  gst_element_add_pad (element, newpad);
  mplex->pads = g_slist_append (mplex->pads, mpad);

  return newpad;
}

static void
gst_mplex_release_pad (GstElement * element, GstPad * pad)
{
 GstMplex *mplex = GST_MPLEX (element);
 GstMplexPad *mpad;

 g_return_if_fail (pad);
 mpad = (GstMplexPad *) gst_pad_get_element_private (pad);
 g_return_if_fail (mpad);

 if (gst_element_remove_pad (element, pad)) {
  gchar *padname;

  GST_MPLEX_MUTEX_LOCK (mplex);
  mpad->eos = TRUE;
  gst_object_unref (mpad->pad);
  mpad->pad = NULL;
  /* wake up if waiting on this pad */
  GST_MPLEX_SIGNAL (mplex, mpad);

  padname = gst_object_get_name (GST_OBJECT (pad));
  if (strstr (padname, "audio")) {
   mplex->num_apads--;
  } else {
   mplex->num_vpads--;
  }
  g_free (padname);

  /* may now be up to us to get things going */
  gst_mplex_start_task (mplex);
  GST_MPLEX_MUTEX_UNLOCK (mplex);
 }
}

static void
gst_mplex_get_property (GObject * object,
  guint prop_id, GValue * value, GParamSpec * pspec)
{
 GST_MPLEX (object)->job->getProperty (prop_id, value);
}
```

```
static void
gst_mplex_set_property (GObject * object,
    guint prop_id, const GValue * value, GParamSpec * pspec)
{
  GST_MPLEX (object)->job->setProperty (prop_id, value);
}

static gboolean
gst_mplex_src_activate_push (GstPad * pad, gboolean active)
{
  gboolean result = TRUE;
  GstMplex *mplex;

  mplex = GST_MPLEX (GST_PAD_PARENT (pad));

  if (active) {
    /* chain will start task once all streams have been setup */
  } else {
    /* end the muxing loop by forcing eos and unblock chains */
    GST_MPLEX_MUTEX_LOCK (mplex);
    mplex->eos = TRUE;
    mplex->srcresult = GST_FLOW_WRONG_STATE;
    GST_MPLEX_SIGNAL_ALL (mplex);
    GST_MPLEX_MUTEX_UNLOCK (mplex);

    /* muxing loop should have ended now and can be joined */
    result = gst_pad_stop_task (pad);
  }

  return result;
}

static GstStateChangeReturn
gst_mplex_change_state (GstElement * element, GstStateChange transition)
{
  GstMplex *mplex = GST_MPLEX (element);
  GstStateChangeReturn ret;

  switch (transition) {
    case GST_STATE_CHANGE_NULL_TO_READY:
      break;
    case GST_STATE_CHANGE_READY_TO_PAUSED:
      break;
    case GST_STATE_CHANGE_PAUSED_TO_PLAYING:
      break;
    default:
      break;
  }
```

```c
  ret = GST_ELEMENT_CLASS (parent_class)->change_state (element, transition);
  if (ret == GST_STATE_CHANGE_FAILURE)
    goto done;

  switch (transition) {
    case GST_STATE_CHANGE_PAUSED_TO_READY:
      gst_mplex_reset (mplex);
      break;
    default:
      break;
  }

done:
  return ret;
}

#ifndef GST_DISABLE_GST_DEBUG

static mjpeg_log_handler_t old_handler = NULL;

/* note that this will affect all mjpegtools elements/threads */
static void
gst_mplex_log_callback (log_level_t level, const char *message)
{
  GstDebugLevel gst_level;

#if GST_MJPEGTOOLS_API >= 10900
  static const gint mjpeg_log_error = mjpeg_loglev_t ("error");
  static const gint mjpeg_log_warn = mjpeg_loglev_t ("warn");
  static const gint mjpeg_log_info = mjpeg_loglev_t ("info");
  static const gint mjpeg_log_debug = mjpeg_loglev_t ("debug");
#else
  static const gint mjpeg_log_error = LOG_ERROR;
  static const gint mjpeg_log_warn = LOG_WARN;
  static const gint mjpeg_log_info = LOG_INFO;
  static const gint mjpeg_log_debug = LOG_DEBUG;
#endif

  if (level == mjpeg_log_error) {
    gst_level = GST_LEVEL_ERROR;
  } else if (level == mjpeg_log_warn) {
    gst_level = GST_LEVEL_WARNING;
  } else if (level == mjpeg_log_info) {
    gst_level = GST_LEVEL_INFO;
  } else if (level == mjpeg_log_debug) {
    gst_level = GST_LEVEL_DEBUG;
  } else {
```

```
    gst_level = GST_LEVEL_INFO;
  }

  /* message could have a % in it, do not segfault in such case */
  gst_debug_log (mplex_debug, gst_level, "", "", 0, NULL, "%s", message);

  /* chain up to the old handler;
   * this could actually be a handler from another mjpegtools based
   * gstreamer element; in which case messages can come out double or from
   * the wrong element ... */
  old_handler (level, message);
}
#endif

static gboolean
plugin_init (GstPlugin * plugin)
{
#ifndef GST_DISABLE_GST_DEBUG
  old_handler = mjpeg_log_set_handler (gst_mplex_log_callback);
  g_assert (old_handler != NULL);
#endif
  /* in any case, we do not want default handler output */
  mjpeg_default_handler_verbosity (0);

  return gst_element_register (plugin, "mplex", GST_RANK_NONE, GST_TYPE_MPLEX);
}

GST_PLUGIN_DEFINE (GST_VERSION_MAJOR,
  GST_VERSION_MINOR,
  "mplex",
  "High-quality MPEG/DVD/SVCD/VCD video/audio multiplexer",
  plugin_init, VERSION, "GPL", GST_PACKAGE_NAME, GST_PACKAGE_ORIGIN)
/* GStreamer mplex (mjpegtools) wrapper
 * (c) 2003 Ronald Bultje <rbultje@ronald.bitfreak.net>
 *
 * gstmplex.hh: gstreamer mplex wrapper
 *
 * 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., 59 Temple Place - Suite 330,
 * Boston, MA 02111-1307, USA.
 */

#ifndef __GST_MPLEX_H__
#define __GST_MPLEX_H__

#include <gst/gst.h>
#include <gst/base/gstadapter.h>
#include <multiplexor.hpp>
#include "gstmplexibitstream.hh"
#include "gstmplexjob.hh"

G_BEGIN_DECLS

#define GST_TYPE_MPLEX \
 (gst_mplex_get_type ())
#define GST_MPLEX(obj) \
 (G_TYPE_CHECK_INSTANCE_CAST ((obj), GST_TYPE_MPLEX, GstMplex))
#define GST_MPLEX_CLASS(klass) \
 (G_TYPE_CHECK_CLASS_CAST ((klass), GST_TYPE_MPLEX, GstMplex))
#define GST_IS_MPLEX(obj) \
 (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GST_TYPE_MPLEX))
#define GST_IS_MPLEX_CLASS(obj) \
 (G_TYPE_CHECK_CLASS_TYPE ((klass), GST_TYPE_MPLEX))

GST_DEBUG_CATEGORY_EXTERN (mplex_debug);
#define GST_CAT_DEFAULT mplex_debug

#define GST_MPLEX_MUTEX_LOCK(m) G_STMT_START {                     \
 GST_LOG_OBJECT (m, "locking tlock from thread %p", g_thread_self ()); \
 g_mutex_lock ((m)->tlock);                                   \
 GST_LOG_OBJECT (m, "locked tlock from thread %p", g_thread_self ()); \
} G_STMT_END

#define GST_MPLEX_MUTEX_UNLOCK(m) G_STMT_START {                     \
 GST_LOG_OBJECT (m, "unlocking tlock from thread %p", g_thread_self ()); \
 g_mutex_unlock ((m)->tlock);                                   \
} G_STMT_END

#define GST_MPLEX_WAIT(m, p) G_STMT_START {                     \
 GST_LOG_OBJECT (m, "thread %p waiting", g_thread_self ());       \
 g_cond_wait ((p)->cond, (m)->tlock);                     \
} G_STMT_END

#define GST_MPLEX_SIGNAL(m, p) G_STMT_START {                     \
 GST_LOG_OBJECT (m, "signalling from thread %p", g_thread_self ());    \
```

```
    g_cond_signal ((p)->cond);                                    \
   } G_STMT_END

   #define GST_MPLEX_SIGNAL_ALL(m) G_STMT_START {                  \
   GST_LOG_OBJECT (m, "signalling all from thread %p", g_thread_self ());   \
   GSList *walk = m->pads;                                         \
   while (walk) {                                                  \
    GST_MPLEX_SIGNAL (m, (GstMplexPad *) walk->data);             \
    walk = walk->next;                                            \
   }                                                               \
   } G_STMT_END

   typedef struct _GstMplexPad
   {
    /* associated pad */
    GstPad *pad;
    /* with mplex TLOCK */
    /* adapter collecting buffers for this pad */
    GstAdapter *adapter;
    /* no more to expect on this pad */
    gboolean eos;
    /* signals counterpart thread to have a look */
    GCond *cond;
    /* amount needed by mplex on this stream */
    guint needed;
    /* bitstream for this pad */
    GstMplexIBitStream *bs;
   } GstMplexPad;

   typedef struct _GstMplex {
    GstElement parent;

    /* pads */
    GSList *pads;
    GstPad *srcpad;
    guint num_apads, num_vpads;

    /* options wrapper */
    GstMplexJob *job;

    /* lock for syncing */
    GMutex *tlock;
    /* with TLOCK */
    /* muxer writer generated eos */
    gboolean eos;
    /* flowreturn obtained by muxer task */
    GstFlowReturn srcresult;
   } GstMplex;
```

```
typedef struct _GstMplexClass {
 GstElementClass parent;
} GstMplexClass;

GType   gst_mplex_get_type   (void);

G_END_DECLS

#endif /* __GST_MPLEX_H__ */
```
/* GStreamer mplex (mjpegtools) wrapper
* (c) 2003 Ronald Bultje <rbultje@ronald.bitfreak.net>
* (c) 2008 Mark Nauwelaerts <mnauw@users.sourceforge.net>
*
* gstmplexibitstream.hh: gstreamer/mplex input bitstream wrapper
*
* 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., 59 Temple Place - Suite 330,
* Boston, MA 02111-1307, USA.
*/

#ifdef HAVE_CONFIG_H
#include "config.h"
#endif

#include <string.h>

#include "gstmplex.hh"
#include "gstmplexibitstream.hh"

/*
* Class init/exit functions.
*/

GstMplexIBitStream::GstMplexIBitStream (GstMplexPad * _data, guint buf_size):
IBitStream ()
{

```
  mpad = _data;
  mplex = GST_MPLEX (GST_PAD_PARENT (mpad->pad));
  eos = FALSE;

  SetBufSize (buf_size);
  eobs = false;
  byteidx = 0;
}

/*
* Read data.
*/

size_t
  GstMplexIBitStream::ReadStreamBytes (uint8_t * buf, size_t size =
  BUFFER_SIZE)
{
  guint8 *data;

  GST_MPLEX_MUTEX_LOCK (mplex);

  GST_DEBUG_OBJECT (mplex, "needing %d bytes", (guint) size);

  while (gst_adapter_available (mpad->adapter) < size
    && !mplex->eos && !mpad->eos) {
    mpad->needed = size;
    GST_MPLEX_SIGNAL (mplex, mpad);
    GST_MPLEX_WAIT (mplex, mpad);
  }

  mpad->needed = 0;
  size = MIN (size, gst_adapter_available (mpad->adapter));
  if (size) {
    data = gst_adapter_take (mpad->adapter, size);
    memcpy (buf, data, size);
    g_free (data);
  }

  GST_MPLEX_MUTEX_UNLOCK (mplex);

  return size;
}

/*
* Are we at EOS?
*/

bool GstMplexIBitStream::EndOfStream (void)
```

```
{
 return eos;
}

bool GstMplexIBitStream::ReadBuffer ()
{
 return ReadIntoBuffer (BUFFER_SIZE);
}
/* GStreamer mplex (mjpegtools) wrapper
 * (c) 2003 Ronald Bultje <rbultje@ronald.bitfreak.net>
 * (c) 2008 Mark Nauwelaerts <mnauw@users.sourceforge.net>
 *
 * gstmplexibitstream.hh: gstreamer/mplex input bitstream wrapper
 *
 * 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., 59 Temple Place - Suite 330,
 * Boston, MA 02111-1307, USA.
 */

#ifndef __GST_MPLEXIBITSTREAM_H__
#define __GST_MPLEXIBITSTREAM_H__

#include <gst/gst.h>
#include <mjpeg_types.h>
#include <bits.hpp>

#include "gstmplex.hh"

/* forward declaration; break circular referencing */
typedef struct _GstMplex GstMplex;
typedef struct _GstMplexPad GstMplexPad;

class GstMplexIBitStream : public IBitStream {
public:
 GstMplexIBitStream (GstMplexPad *pad, guint buf_size = BUFFER_SIZE);
 bool ReadBuffer ();
```

```
protected:
 /* read data */
 size_t ReadStreamBytes (uint8_t *buf, size_t number);

 /* are we at EOS? */
 bool EndOfStream (void);

private:
 GstMplex *mplex;
 GstMplexPad *mpad;
 gboolean eos;
};

#endif /* __GST_MPLEXIBITSTREAM_H__ */
/* GStreamer mplex (mjpegtools) wrapper
* (c) 2003 Ronald Bultje <rbultje@ronald.bitfreak.net>
*
* gstmplexjob.hh: gstreamer/mplex multiplex-job wrapper
*
* 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., 59 Temple Place - Suite 330,
* Boston, MA 02111-1307, USA.
*/

#ifdef HAVE_CONFIG_H
#include "config.h"
#endif

#include "gstmplexjob.hh"


enum
{
 ARG_0,
 ARG_FORMAT,
 ARG_MUX_BITRATE,
 ARG_VBR,
```

```
  ARG_SYSTEM_HEADERS,
  ARG_SPLIT_SEQUENCE,
  ARG_SEGMENT_SIZE,
  ARG_PACKETS_PER_PACK,
  ARG_SECTOR_SIZE,
  ARG_BUFSIZE
    /* FILL ME */
};

/*
 * Property enumeration types.
 */

#define GST_TYPE_MPLEX_FORMAT \
 (gst_mplex_format_get_type ())

static GType
gst_mplex_format_get_type (void)
{
 static GType mplex_format_type = 0;

 if (!mplex_format_type) {
   static const GEnumValue mplex_formats[] = {
     {0, "Generic MPEG-1", "0"},
     {1, "Standard VCD", "1"},
     {2, "User VCD", "2"},
     {3, "Generic MPEG-2", "3"},
     {4, "Standard SVCD", "4"},
     {5, "User SVCD", "5"},
     {6, "VCD Stills sequences", "6"},
     {7, "SVCD Stills sequences", "7"},
     {8, "DVD MPEG-2 for dvdauthor", "8"},
     {9, "DVD MPEG-2", "9"},
     {0, NULL, NULL},
   };

   mplex_format_type =
     g_enum_register_static ("GstMplexFormat", mplex_formats);
 }

 return mplex_format_type;
}

/*
 * Class init functions.
 */

GstMplexJob::GstMplexJob (void):
```

```
MultiplexJob ()
{
 /* blabla */
 bufsize = 0;
}

/*
 * GObject properties.
 */

void
GstMplexJob::initProperties (GObjectClass * klass)
{
 /* encoding profile */
 g_object_class_install_property (klass, ARG_FORMAT,
    g_param_spec_enum ("format", "Format", "Encoding profile format",
       GST_TYPE_MPLEX_FORMAT, 0,
       (GParamFlags) (G_PARAM_READWRITE | G_PARAM_STATIC_STRINGS)));

 /* total stream datarate. Normally, this shouldn't be needed, but
  * some DVD/VCD/SVCD players really need strict values to handle
  * the created files correctly. */
 g_object_class_install_property (klass, ARG_MUX_BITRATE,
    g_param_spec_int ("mux-bitrate", "Mux. bitrate",
       "Bitrate of output stream in kbps (0 = autodetect)",
       0, 15 * 1024, 0,
       (GParamFlags) (G_PARAM_READWRITE | G_PARAM_STATIC_STRINGS)));

 /* override decode buffer size otherwise determined by format */
 g_object_class_install_property (klass, ARG_BUFSIZE,
    g_param_spec_int ("bufsize", "Decoder buf. size",
       "Target decoders video buffer size (kB) "
       "[default determined by format if not explicitly set]",
       20, 4000, 46,
       (GParamFlags) (G_PARAM_READWRITE | G_PARAM_STATIC_STRINGS)));

 /* some boolean stuff for headers */
 g_object_class_install_property (klass, ARG_VBR,
    g_param_spec_boolean ("vbr", "VBR",
       "Whether the input video stream is variable bitrate",
       FALSE,
       (GParamFlags) (G_PARAM_READWRITE | G_PARAM_STATIC_STRINGS)));
 g_object_class_install_property (klass, ARG_SYSTEM_HEADERS,
    g_param_spec_boolean ("system-headers", "System headers",
       "Create system header in every pack for generic formats",
       FALSE,
       (GParamFlags) (G_PARAM_READWRITE | G_PARAM_STATIC_STRINGS)));
#if 0                    /* not supported */
```

```c
  g_object_class_install_property (klass, ARG_SPLIT_SEQUENCE,
     g_param_spec_boolean ("split-sequence", "Split sequence",
        "Simply split a sequence across files "
        "(rather than building run-out/run-in)",
        FALSE,
        (GParamFlags) (G_PARAM_READWRITE | G_PARAM_STATIC_STRINGS)));


  /* size of a segment */
  g_object_class_install_property (klass, ARG_SEGMENT_SIZE,
     g_param_spec_int ("max-segment-size", "Max. segment size",
        "Max. size per segment/file in MB (0 = unlimited)",
        0, 10 * 1024, 0,
        (GParamFlags) (G_PARAM_READWRITE | G_PARAM_STATIC_STRINGS)));
#endif


  /* packets per pack (generic formats) */
  g_object_class_install_property (klass, ARG_PACKETS_PER_PACK,
     g_param_spec_int ("packets-per-pack", "Packets per pack",
        "Number of packets per pack for generic formats",
        1, 100, 1,
        (GParamFlags) (G_PARAM_READWRITE | G_PARAM_STATIC_STRINGS)));


  /* size of one sector */
  g_object_class_install_property (klass, ARG_SECTOR_SIZE,
     g_param_spec_int ("sector-size", "Sector size",
        "Specify sector size in bytes for generic formats",
        256, 16384, 2048,
        (GParamFlags) (G_PARAM_READWRITE | G_PARAM_STATIC_STRINGS)));
}

/*
 * set/get gobject properties.
 */

void
GstMplexJob::getProperty (guint prop_id, GValue * value)
{
 switch (prop_id) {
   case ARG_FORMAT:
     g_value_set_enum (value, mux_format);
     break;
   case ARG_MUX_BITRATE:
     /* convert from bytes back to bits */
     g_value_set_int (value, (data_rate * 8) / 1000);
     break;
   case ARG_VBR:
     g_value_set_boolean (value, VBR);
     break;
```

```
    case ARG_SYSTEM_HEADERS:
      g_value_set_boolean (value, always_system_headers);
      break;
    case ARG_SPLIT_SEQUENCE:
      g_value_set_boolean (value, multifile_segment);
      break;
    case ARG_SEGMENT_SIZE:
      g_value_set_int (value, max_segment_size);
      break;
    case ARG_PACKETS_PER_PACK:
      g_value_set_int (value, packets_per_pack);
      break;
    case ARG_SECTOR_SIZE:
      g_value_set_int (value, sector_size);
      break;
    case ARG_BUFSIZE:
      g_value_set_int (value, bufsize);
      break;
    default:
      break;
  }
}

void
GstMplexJob::setProperty (guint prop_id, const GValue * value)
{
  switch (prop_id) {
    case ARG_FORMAT:
      mux_format = g_value_get_enum (value);
      break;
    case ARG_MUX_BITRATE:
      /* data_rate expects bytes (don't ask me why the property itself is
       * in bits, I'm just staying compatible to mjpegtools options), and
       * rounded up to 50-bytes. */
      data_rate = ((g_value_get_int (value) * 1000 / 8 + 49) / 50) * 50;
      break;
    case ARG_VBR:
      VBR = g_value_get_boolean (value);
      break;
    case ARG_SYSTEM_HEADERS:
      always_system_headers = g_value_get_boolean (value);
      break;
    case ARG_SPLIT_SEQUENCE:
      multifile_segment = g_value_get_boolean (value);
      break;
    case ARG_SEGMENT_SIZE:
      max_segment_size = g_value_get_int (value);
      break;
```

```
    case ARG_PACKETS_PER_PACK:
      packets_per_pack = g_value_get_int (value);
      break;
    case ARG_SECTOR_SIZE:
      sector_size = g_value_get_int (value);
      break;
    case ARG_BUFSIZE:
      bufsize = g_value_get_int (value);
      break;
    default:
      break;
  }
}
/* GStreamer mplex (mjpegtools) wrapper
 * (c) 2003 Ronald Bultje <rbultje@ronald.bitfreak.net>
 *
 * gstmplexjob.hh: gstreamer/mplex multiplex-job wrapper
 *
 * 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., 59 Temple Place - Suite 330,
 * Boston, MA 02111-1307, USA.
 */

#ifndef __GST_MPLEXJOB_H__
#define __GST_MPLEXJOB_H__

#include <glib-object.h>
#include <interact.hpp>

class GstMplexJob : public MultiplexJob {
public:
 GstMplexJob (void);

 /* gobject properties */
 static void initProperties (GObjectClass *klass);

 /* set/get gobject properties */
```

```
void getProperty (guint        prop_id,
    GValue      *value);
void setProperty (guint        prop_id,
    const GValue *value);


int bufsize;
};


#endif /* __GST_MPLEXJOB_H__ */
/* GStreamer mplex (mjpegtools) wrapper
* (c) 2003 Ronald Bultje <rbultje@ronald.bitfreak.net>
* (c) 2008 Mark Nauwelaerts <mnauw@users.sourceforge.net>
*
* gstmplexoutputstream.hh: gstreamer/mplex output stream wrapper
*
* 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., 59 Temple Place - Suite 330,
* Boston, MA 02111-1307, USA.
*/


#ifdef HAVE_CONFIG_H
#include "config.h"
#endif


#include <string.h>


#include "gstmplex.hh"
#include "gstmplexoutputstream.hh"


/*
* Class init functions.
*/


GstMplexOutputStream::GstMplexOutputStream (GstMplex * _element, GstPad * _pad):
OutputStream ()
{
 mplex = _element;
```

```
  pad = _pad;
  size = 0;
}

/*
* Open/close. Basically 'no-op's (close() sets EOS).
*
* Open (): -1 means failure, 0 means success.
*/

int
GstMplexOutputStream::Open (void)
{
  return 0;
}

void
GstMplexOutputStream::Close (void)
{
  GST_MPLEX_MUTEX_LOCK (mplex);
  GST_DEBUG_OBJECT (mplex, "closing stream and sending eos");
  gst_pad_push_event (pad, gst_event_new_eos ());
  /* notify chain there is no more need to supply buffers */
  mplex->eos = TRUE;
  GST_MPLEX_SIGNAL_ALL (mplex);
  GST_MPLEX_MUTEX_UNLOCK (mplex);
}

/*
* Get size of current segment.
*/

#if GST_MJPEGTOOLS_API >= 10900
uint64_t
GstMplexOutputStream::SegmentSize (void)
#else
off_t
GstMplexOutputStream::SegmentSize (void)
#endif
{
  return size;
}

/*
* Next segment; not really supported.
*/

void
```

```
GstMplexOutputStream::NextSegment (void)
{
 size = 0;

 GST_WARNING_OBJECT (mplex, "multiple file output is not supported");
 /* FIXME: no such filesink behaviour to be expected */
}

/*
* Write data.
*/

void
GstMplexOutputStream::Write (guint8 * data, guint len)
{
 GstBuffer *buf;

 buf = gst_buffer_new_and_alloc (len);
 memcpy (GST_BUFFER_DATA (buf), data, len);

 size += len;
 GST_MPLEX_MUTEX_LOCK (mplex);
 gst_buffer_set_caps (buf, GST_PAD_CAPS (pad));
 mplex->srcresult = gst_pad_push (pad, buf);
 GST_MPLEX_MUTEX_UNLOCK (mplex);
}
/* GStreamer mplex (mjpegtools) wrapper
* (c) 2003 Ronald Bultje <rbultje@ronald.bitfreak.net>
* (c) 2008 Mark Nauwelaerts <mnauw@users.sourceforge.net>
*
* gstmplexoutputstream.hh: gstreamer/mplex output stream wrapper
*
* 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., 59 Temple Place - Suite 330,
* Boston, MA 02111-1307, USA.
*/
```

```
#ifndef __GST_MPLEXOUTPUTSTREAM_H__
#define __GST_MPLEXOUTPUTSTREAM_H__

#include <gst/gst.h>
#include <mjpeg_types.h>
#include <outputstrm.hpp>

#include "gstmplex.hh"

class GstMplexOutputStream : public OutputStream {
public:
 GstMplexOutputStream (GstMplex *element, GstPad *pad);

 /* open/close. Basically 'no-op's (close() sets EOS). */
 int  Open  (void);
 void Close (void);

 /* get size of current segment */
#if GST_MJPEGTOOLS_API >= 10900
 uint64_t SegmentSize (void);
#else
 off_t SegmentSize (void);
#endif

 /* next segment */
 void NextSegment (void);

 /* write data */
 void Write (guint8 *data, guint len);

private:
 GstMplex *mplex;
 GstPad *pad;
 guint64 size;
};

#endif /* __GST_MPLEXOUTPUTSTREAM_H__ */
```

1.7 gstreamer-plugins-base 0.10.35.1

1.7.1 Available under license :

GNU LIBRARY GENERAL PUBLIC LICENSE
       Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
          675 Mass Ave, Cambridge, MA 02139, USA
Everyone is permitted to copy and distribute verbatim copies

of this license document, but changing it is not allowed.

[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!
    GNU GENERAL PUBLIC LICENSE
       Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
            59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Library General Public License instead.) You can apply it to
your programs, 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 software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, 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 redistributors of a free
program will individually obtain patent licenses, in effect making the
program proprietary. To prevent this, we have made it clear that any

patent must be licensed for everyone's free use or not licensed at all.

 The precise terms and conditions for copying, distribution and
modification follow.

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

 0. This License applies to any program or other work which contains
a notice placed by the copyright holder saying it may be distributed
under the terms of this General Public License. The "Program", below,
refers to any such program or work, and a "work based on the Program"
means either the Program or any derivative work under copyright law:
that is to say, a work containing the Program or a portion of it,
either verbatim or with modifications and/or translated into another
language. (Hereinafter, translation is included without limitation in
the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not
covered by this License; they are outside its scope. The act of
running the Program is not restricted, and the output from the Program
is covered only if its contents constitute a work based on the
Program (independent of having been made by running the Program).
Whether that is true depends on what the Program does.

 1. You may copy and distribute verbatim copies of the Program's
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 give any other recipients of the Program a copy of this License
along with the Program.

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 Program or any portion
of it, thus forming a work based on the Program, 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) You must cause the modified files to carry prominent notices
  stating that you changed the files and the date of any change.

  b) You must cause any work that you distribute or publish, that in
  whole or in part contains or is derived from the Program or any
  part thereof, to be licensed as a whole at no charge to all third
  parties under the terms of this License.

c) If the modified program normally reads commands interactively
when run, you must cause it, when started running for such
interactive use in the most ordinary way, to print or display an
announcement including an appropriate copyright notice and a
notice that there is no warranty (or else, saying that you provide
a warranty) and that users may redistribute the program under
these conditions, and telling the user how to view a copy of this
License. (Exception: if the Program itself is interactive but
does not normally print such an announcement, your work based on
the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If
identifiable sections of that work are not derived from the Program,
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 Program, 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 Program.

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

 3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of
Sections 1 and 2 above provided that you also do one of the following:

  a) 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; or,

  b) Accompany it with a written offer, valid for at least three
  years, to give any third party, for a charge no more than your
  cost of physically performing source distribution, a complete
  machine-readable copy of the corresponding source code, to be
  distributed under the terms of Sections 1 and 2 above on a medium
  customarily used for software interchange; or,

  c) Accompany it with the information you received as to the offer

to distribute corresponding source code. (This alternative is
allowed only for noncommercial distribution and only if you
received the program in object code or executable form with such
an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, 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 executable. 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.

If distribution of executable or 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 counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

 4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License. Any attempt
otherwise to copy, modify, sublicense or distribute the Program 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.

 5. 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 Program or its derivative works. These actions are
prohibited by law if you do not accept this License. Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

 6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program 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.

 7. 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 Program at all. For example, if a patent
license would not permit royalty-free redistribution of the Program 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 Program.

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.

 8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program 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.

 9. The Free Software Foundation may publish revised and/or new versions
of the 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 Program
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 Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

  10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, 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

  11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "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 PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

  12. 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 PROGRAM 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 PROGRAM (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 PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), 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 Programs

 If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

 To do so, attach the following notices to the program. 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.

Copyright (C) 19yy <name of author>

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

This program 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 General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

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

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) 19yy name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

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

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Library General
Public License instead of this License.

Disclaimer: The GStreamer team makes a real effort to make sure that
the information in this file is as up-to-date and accurate as possible.
However, this file may contain errors and omissions. Some dependant
libraries change their licensing from version-to-version and some of
the libraries listed here as LGPL were under the GPL license in
older releases. Distributions which include GStreamer are responsible
for ensuring that plugins are used in accordance with licensing terms
and other laws.

GStreamer is developed under the terms of the LGPL (see LICENSE file for details).
Some of our plugins however rely on libraries which are available under other
licenses. This means that if you are using an application which has a non-GPL
compatible license, for instance a closed-source application with GStreamer
you have to make sure not to use GPL linked or derived plugins. When using
GPL linked plugins GStreamer is for all practical reasons under the GPL itself.

The plugins which use a GPL library are as follows:

a52dec   liba52   (http://liba52.sourceforge.net/)
aasink   aalib   (http://sourceforge.net/projects/aa-project/)
cdparanoia   libcdparanoia   (http://www.xiph.org/paranoia/)
dtsdec   libdts   (http://www.videolan.org/dtsdec.html)
dvdnavsrc   libdvdnav   (http://dvd.sourceforge.net/)
dxr3   (http://dxr3.sourceforge.net/)
icecastsend   libshout   (http://www.icecast.org)
mad   libmad   (http://www.mars.org/home/rob/proj/mpeg/)
mpeg2dec   libmpeg2   (http://libmpeg2.sourceforge.net/)
mpeg2enc   libmpeg2enc   (http://mjpeg.sourceforge.net/)
mplex   libmplex   (http://mjpeg.sourceforge.net/)

siddec   libsidplay 1.36 (http://www.geocities.com/SiliconValley/Lakes/5147/sidplay/,
                http://sourceforge.net/projects/sidplay2/)

trm   libmusicbrainz (http://www.musicbrainz.org/)

xine   libxine   (http://xinehq.de/)

xvid   libxvidcore   (http://www.xvid.org/)

Plugins derived from GPL code are as follows:

dvdreadsrc   libdvdread   (http://www.dtek.chalmers.se/groups/dvd/)

jack   libjack   (http://jackit.sourceforge.net/)
                Note libjack is LGPL, but plugin is GPL.

monoscope   None   (Algorithm by Ralph Loader, Joerg Walter,
                Richard Boulton, and Andy Lo A Foe)

rtjpeg   None   (Erik Walthinsen's algorithm)

rtp   None   (http://www.linphone.org/ortp/)

synaesthesia   (http://www.logarithmic.net/pfh/synaesthesia)

system_encode None   (Algorithm by Chrisoph Moar, Wim Tayman's and
                Erik Walthinsen)

vbidec   None   (Algorithm by Billy Biggs, Doug Bell,
                Erik Walthinsen and David I. Lehn)

Plugins which use a LGPL library are as follows:

alsa   alsa   (http://alsa-project.org/)

artsdsink   aRts   (http://arts-project.org/)

cacasink   libcaca   (http://sam.zoy.org/projects/libcaca/)

colorspacelcs   libcolorspace

dvdec   libdv   (http://libdv.sourceforge.net/)

esdmon,esdsink libesd   (ftp.gnome.org/pub/GNOME/stable/sources/esound/)

faac   libfaac   (http://www.audiocoding.com/modules/mydownloads/)

fameenc   libfame   (http://fame.sourceforge.net/)

ffmpeg   ffmpeg   (http://ffmpeg.sourceforge.net/)

gdkpixbuf   GTK+   (http://www.gtk.org/)

gnomevfs   gnome-vfs   (ftp.gnome.org//pub/GNOME/stable/sources/gnome-vfs/)

gst_arts   aRts   (http://arts-project.org/)

gst1394   libraw1394   (http://www.linux1394.org/)

gstaf   libaudiofile   (http://www.68k.org/~michael/audiofile/)

gstsf   libsndfile   (http://www.mega-nerd.com/libsndfile/)

hermescolorspace Hermes   (http://www.clanlib.org/hermes/)

kio   KDE   (http://www.kde.org/)

ladspa   (http://www.ladspa.org/)

lame   libmp3lame   (http://lame.sourceforge.net/)

libvisual   libvisual   (http://libvisual.sourceforge.net/)

matroska   (http://www.matroska.org/)

mikmod   libmikmod   (http://mikmod.raphnet.net/)

ossaudio   (http://www.opensound.com/)

qcamsrc

rfbenc   librfb

sdlvideosink   libsdl   (http://www.libsdl.org/)
shout2send   libshout2   (http://www.icecast.org)
smoothwave
swfdec   (http://swfdec.sourceforge.net/)
tarkin   (http://svn.xiph.org/trunk/tarkin/)
textoverlay   pango   (http://www.pango.org/)
dirac   (http://www.bbc.co.uk/rd/projects/dirac/)
effectv (Our ports was relicensed)(http://effectv.sourceforge.net/)
musepack   (http://www.musepack.net/)

Plugins which use a BSD covered library are as follows:

ogg   libogg/libvorbis (http://www.xiph.org/ogg/vorbis/)
vorbis   libogg/libvorbis (http://www.xiph.org/ogg/vorbis/)
gsttheora   libtheora   (http://www.theora.org/)
speex   (http://www.speex.org/)
flac   libFLAC   (http://flac.sourceforge.net/)

Plugins based on libraries with other free licenses:

ximagesink   libXv
xvimagesink   libXv
- license: MIT X11 / X Consortium license

gsm   libgsm
- license   MIT license http://kbs.cs.tu-berlin.de/~jutta/toast.html

festival   (http://www.cstr.ed.ac.uk/projects/festival/)
- license: http://www.cstr.ed.ac.uk/projects/festival/freecopyright.shtml

jpeg   (http://www.ijg.org/)
- license: IJG license

nas   (http://radscan.com/nas.html)
- license: NAS license

snapshot   libpng   (http://www.libpng.org/pub/png/)
- license: PNG license

mngdec/mngenc libmng   (http://gjuyn.xs4all.nl/libmng)
- license   PNG license

Plugins using non-free libraries:
divxdec, divxenc   (http://www.divx.com/)
osxaudio   (http://www.apple.com/macosx/)
sunaudiosink   (http://www.sun.com/)

Unsure:

faad libfaad (http://www.audiocoding.com/)

faad's license depends on its major version :

 - for FAAD1 : it is LGPL

 - for FAAD2 : it is GPL

look at these pages for more information on licenses :

http://www.audiocoding.com/modules/wiki/?page=FAAD2

http://www.audiocoding.com/modules/wiki/?page=FAAC

by default, GStreamer faad plugin would compile against FAAD2 if available.

it falls back to FAAD1 if not.

1.8 gstreamer-plugins-good 0.10.30.1

1.8.1 Available under license :

GNU LESSER GENERAL PUBLIC LICENSE

 Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.

 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies

of this license document, but changing it is not allowed.

[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 St, 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!
/* GStreamer
* Copyright (C) <1999> Erik Walthinsen <omega@cse.ogi.edu>
*
* EffecTV:
* Copyright (C) 2001 FUKUCHI Kentarou
*
* EffecTV is free software. We release this product under the terms of the
* GNU General Public License version 2. The license is included in the file
* COPYING.
*

* This program 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 General Public License for more details.
*/
This is a demuxer supporting a subset of the Quicktime video container
format developed by Apple. Apple and others have some patents on
some features of the Quicktime container format in regards to technologies
such as QuicktimeVR and RTP hinting. Due to that be aware that if ever
such features are added to this demuxer it would need to be moved to the
-ugly module or those features need to come as add-in functionality stored in
another module.

As the plugin is as of today's date (19th of June 2007) it does not
violate any software patents we know of.
/*
* Unless otherwise indicated, Source Code is licensed under MIT license.
*
* Permission is hereby granted, free of charge, to any person obtaining a copy of
* this software and associated documentation files (the "Software"), to deal in
* the Software without restriction, including without limitation the rights to
* use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
* of the Software, and to permit persons to whom the Software is furnished to do
* so, subject to the following conditions:
*
* The above copyright notice and this permission notice shall be included in all
* copies or substantial portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
* AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
* LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
* OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
* SOFTWARE.
*/

Disclaimer: The GStreamer team makes a real effort to make sure that
the information in this file is as up-to-date and accurate as possible.
However, this file may contain errors and omissions. Some dependant
libraries change their licensing from version-to-version and some of
the libraries listed here as LGPL were under the GPL license in
older releases. Distributions which include GStreamer are responsible
for ensuring that plugins are used in accordance with licensing terms
and other laws.

GStreamer is developed under the terms of the LGPL (see LICENSE file for details).
Some of our plugins however rely on libraries which are available under other
licenses. This means that if you are using an application which has a non-GPL

compatible license, for instance a closed-source application with GStreamer
you have to make sure not to use GPL linked or derived plugins. When using
GPL linked plugins GStreamer is for all practical reasons under the GPL itself.

The plugins which use a GPL library are as follows:

a52dec   liba52   (http://liba52.sourceforge.net/)
aasink   aalib   (http://sourceforge.net/projects/aa-project/)
cdparanoia   libcdparanoia   (http://www.xiph.org/paranoia/)
dtsdec   libdts   (http://www.videolan.org/dtsdec.html)
dvdnavsrc   libdvdnav   (http://dvd.sourceforge.net/)
dxr3   (http://dxr3.sourceforge.net/)
icecastsend   libshout   (http://www.icecast.org)
mad   libmad   (http://www.mars.org/home/rob/proj/mpeg/)
mpeg2dec   libmpeg2   (http://libmpeg2.sourceforge.net/)
mpeg2enc   libmpeg2enc   (http://mjpeg.sourceforge.net/)
mplex   libmplex   (http://mjpeg.sourceforge.net/)
siddec   libsidplay 1.36 (http://www.geocities.com/SiliconValley/Lakes/5147/sidplay/,
                                http://sourceforge.net/projects/sidplay2/)
trm   libmusicbrainz (http://www.musicbrainz.org/)
xine   libxine   (http://xinehq.de/)
xvid   libxvidcore   (http://www.xvid.org/)

Plugins derived from GPL code are as follows:

dvdreadsrc   libdvdread   (http://www.dtek.chalmers.se/groups/dvd/)
jack   libjack   (http://jackit.sourceforge.net/)
                     Note libjack is LGPL, but plugin is GPL.
monoscope   None   (Algorithm by Ralph Loader, Joerg Walter,
                    Richard Boulton, and Andy Lo A Foe)
rtjpeg   None   (Erik Walthinsen's algorithm)
rtp   None   (http://www.linphone.org/ortp/)
synaesthesia   (http://www.logarithmic.net/pfh/synaesthesia)
system_encode   None   (Algorithm by Chrisoph Moar, Wim Tayman's and
                    Erik Walthinsen)
vbidec   None   (Algorithm by Billy Biggs, Doug Bell,
                    Erik Walthinsen and David I. Lehn)

Plugins which use a LGPL library are as follows:

alsa   alsa   (http://alsa-project.org/)
artsdsink   aRts   (http://arts-project.org/)
cacasink   libcaca   (http://sam.zoy.org/projects/libcaca/)
colorspacelcs   libcolorspace
dvdec   libdv   (http://libdv.sourceforge.net/)
esdmon,esdsink libesd   (ftp.gnome.org/pub/GNOME/stable/sources/esound/)
faac   libfaac   (http://www.audiocoding.com/modules/mydownloads/)
fameenc   libfame   (http://fame.sourceforge.net/)

ffmpeg   ffmpeg   (http://ffmpeg.sourceforge.net/)

gdkpixbuf   GTK+   (http://www.gtk.org/)

gnomevfs   gnome-vfs   (ftp.gnome.org//pub/GNOME/stable/sources/gnome-vfs/)

gst_arts   aRts   (http://arts-project.org/)

gst1394   libraw1394   (http://www.linux1394.org/)

gstaf   libaudiofile   (http://www.68k.org/~michael/audiofile/)

gstsf   libsndfile   (http://www.mega-nerd.com/libsndfile/)

hermescolorspace Hermes   (http://www.clanlib.org/hermes/)

kio   KDE   (http://www.kde.org/)

ladspa   (http://www.ladspa.org/)

lame   libmp3lame   (http://lame.sourceforge.net/)

libvisual   libvisual   (http://libvisual.sourceforge.net/)

matroska   (http://www.matroska.org/)

mikmod   libmikmod   (http://mikmod.raphnet.net/)

ossaudio   (http://www.opensound.com/)

qcamsrc

rfbenc   librfb

sdlvideosink   libsdl   (http://www.libsdl.org/)

shout2send   libshout2   (http://www.icecast.org)

smoothwave

swfdec   (http://swfdec.sourceforge.net/)

tarkin   (http://svn.xiph.org/trunk/tarkin/)

textoverlay   pango   (http://www.pango.org/)

dirac   (http://www.bbc.co.uk/rd/projects/dirac/)

effectv (Our ports was relicensed)(http://effectv.sourceforge.net/)

musepack   (http://www.musepack.net/)

Plugins which use a BSD covered library are as follows:

ogg   libogg/libvorbis (http://www.xiph.org/ogg/vorbis/)

vorbis   libogg/libvorbis (http://www.xiph.org/ogg/vorbis/)

gsttheora   libtheora   (http://www.theora.org/)

speex   (http://www.speex.org/)

flac   libFLAC   (http://flac.sourceforge.net/)

Plugins based on libraries with other free licenses:

ximagesink   libXv

xvimagesink   libXv

- license: MIT X11 / X Consortium license

gsm   libgsm

- license   MIT license http://kbs.cs.tu-berlin.de/~jutta/toast.html

festival   (http://www.cstr.ed.ac.uk/projects/festival/)

- license: http://www.cstr.ed.ac.uk/projects/festival/freecopyright.shtml

jpeg   (http://www.ijg.org/)

- license: IJG license

nas   (http://radscan.com/nas.html)
- license: NAS license

snapshot   libpng   (http://www.libpng.org/pub/png/)
- license: PNG license

mngdec/mngenc libmng   (http://gjuyn.xs4all.nl/libmng)
- license   PNG license

Plugins using non-free libraries:
divxdec, divxenc   (http://www.divx.com/)
osxaudio   (http://www.apple.com/macosx/)
sunaudiosink   (http://www.sun.com/)

Unsure:

faad   libfaad   (http://www.audiocoding.com/)
faad's license depends on its major version :
 - for FAAD1 : it is LGPL
 - for FAAD2 : it is GPL
look at these pages for more information on licenses :
http://www.audiocoding.com/modules/wiki/?page=FAAD2
http://www.audiocoding.com/modules/wiki/?page=FAAC

by default, GStreamer faad plugin would compile against FAAD2 if available.
it falls back to FAAD1 if not.
/*
* GStreamer
* Copyright (C) 2007 Sebastian Drge <slomo@circular-chaos.org>
* Copyright (C) 2006 Stefan Kost <ensonic@users.sf.net>
*
* 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., 59 Temple Place - Suite 330,
* Boston, MA 02111-1307, USA.
*/

```
/**
 * SECTION:element-audioamplify
 *
 * Amplifies an audio stream by a given factor and allows the selection of different clipping modes.
 * The difference between the clipping modes is best evaluated by testing.
 *
 * <refsect2>
 * <title>Example launch line</title>
 * |[
 * gst-launch audiotestsrc wave=saw ! audioamplify amplification=1.5 ! alsasink
 * gst-launch filesrc location="melo1.ogg" ! oggdemux ! vorbisdec ! audioconvert ! audioamplify amplification=1.5
method=wrap-negative ! alsasink
 * gst-launch audiotestsrc wave=saw ! audioconvert ! audioamplify amplification=1.5 method=wrap-positive !
audioconvert ! alsasink
 * ]|
 * </refsect2>
 */

#ifdef HAVE_CONFIG_H
#include "config.h"
#endif

#include <gst/gst.h>
#include <gst/base/gstbasetransform.h>
#include <gst/audio/audio.h>
#include <gst/audio/gstaudiofilter.h>
#include <gst/controller/gstcontroller.h>

#include "audioamplify.h"

#define GST_CAT_DEFAULT gst_audio_amplify_debug
GST_DEBUG_CATEGORY_STATIC (GST_CAT_DEFAULT);

/* Filter signals and args */
enum
{
 /* FILL ME */
 LAST_SIGNAL
};

enum
{
 PROP_0,
 PROP_AMPLIFICATION,
 PROP_CLIPPING_METHOD
};
```

```c
enum
{
 METHOD_CLIP = 0,
 METHOD_WRAP_NEGATIVE,
 METHOD_WRAP_POSITIVE,
 METHOD_NOCLIP,
 NUM_METHODS
};

#define GST_TYPE_AUDIO_AMPLIFY_CLIPPING_METHOD (gst_audio_amplify_clipping_method_get_type
())
static GType
gst_audio_amplify_clipping_method_get_type (void)
{
 static GType gtype = 0;

 if (gtype == 0) {
   static const GEnumValue values[] = {
     {METHOD_CLIP, "Normal clipping (default)", "clip"},
     {METHOD_WRAP_NEGATIVE,
         "Push overdriven values back from the opposite side",
       "wrap-negative"},
     {METHOD_WRAP_POSITIVE, "Push overdriven values back from the same side",
       "wrap-positive"},
     {METHOD_NOCLIP, "No clipping", "none"},
     {0, NULL, NULL}
   };

   /* FIXME 0.11: rename to GstAudioAmplifyClippingMethod */
   gtype = g_enum_register_static ("GstAudioPanoramaClippingMethod", values);
 }
 return gtype;
}

#define ALLOWED_CAPS                               \
   "audio/x-raw-int,"                              \
   " depth=(int)8,"                                \
   " width=(int)8,"                                \
   " endianness=(int)BYTE_ORDER,"                  \
   " signed=(bool)TRUE,"                           \
   " rate=(int)[1,MAX],"                           \
   " channels=(int)[1,MAX]; "                      \
   "audio/x-raw-int,"                              \
   " depth=(int)16,"                               \
   " width=(int)16,"                               \
   " endianness=(int)BYTE_ORDER,"                  \
   " signed=(bool)TRUE,"                           \
   " rate=(int)[1,MAX],"                           \
```

```
  " channels=(int)[1,MAX]; "                        \
  "audio/x-raw-int,"                           \
  " depth=(int)32,"                           \
  " width=(int)32,"                           \
  " endianness=(int)BYTE_ORDER,"                     \
  " signed=(bool)TRUE,"                         \
  " rate=(int)[1,MAX],"                         \
  " channels=(int)[1,MAX]; "                       \
  "audio/x-raw-float,"                         \
  " width=(int){32,64},"                        \
  " endianness=(int)BYTE_ORDER,"                     \
  " rate=(int)[1,MAX],"                         \
  " channels=(int)[1,MAX]"


#define DEBUG_INIT(bla) \
 GST_DEBUG_CATEGORY_INIT (gst_audio_amplify_debug, "audioamplify", 0, "audioamplify element");


GST_BOILERPLATE_FULL (GstAudioAmplify, gst_audio_amplify, GstAudioFilter,
  GST_TYPE_AUDIO_FILTER, DEBUG_INIT);


static gboolean gst_audio_amplify_set_process_function (GstAudioAmplify *
  filter, gint clipping, gint format, gint width);
static void gst_audio_amplify_set_property (GObject * object, guint prop_id,
  const GValue * value, GParamSpec * pspec);
static void gst_audio_amplify_get_property (GObject * object, guint prop_id,
  GValue * value, GParamSpec * pspec);


static gboolean gst_audio_amplify_setup (GstAudioFilter * filter,
  GstRingBufferSpec * format);
static GstFlowReturn gst_audio_amplify_transform_ip (GstBaseTransform * base,
  GstBuffer * buf);


#define MIN_gint8 G_MININT8
#define MAX_gint8 G_MAXINT8
#define MIN_gint16 G_MININT16
#define MAX_gint16 G_MAXINT16
#define MIN_gint32 G_MININT32
#define MAX_gint32 G_MAXINT32


#define MAKE_INT_FUNCS(type,largetype)                   \
static void                              \
gst_audio_amplify_transform_##type##_clip (GstAudioAmplify * filter,     \
  void * data, guint num_samples)                    \
{                                  \
 type *d = data;                          \
                                  \
 while (num_samples--) {                       \
  largetype val = *d * filter->amplification;              \
```

```
    *d++ = CLAMP (val, MIN_##type, MAX_##type);                    \
  }                                                                \
  }                                                                \
  static void                                                      \
  gst_audio_amplify_transform_##type##_wrap_negative (GstAudioAmplify * filter, \
    void * data, guint num_samples)                                \
  {                                                                \
   type *d = data;                                                 \
                                                                   \
   while (num_samples--) {                                         \
    largetype val = *d * filter->amplification;                    \
    if (val > MAX_##type)                                          \
      val = MIN_##type + (val - MIN_##type) % ((largetype) MAX_##type + 1 -  \
        MIN_##type);                                               \
    else if (val < MIN_##type)                                     \
      val = MAX_##type - (MAX_##type - val) % ((largetype) MAX_##type + 1 -  \
        MIN_##type);                                               \
    *d++ = val;                                                    \
  }                                                                \
  }                                                                \
  static void                                                      \
  gst_audio_amplify_transform_##type##_wrap_positive (GstAudioAmplify * filter, \
    void * data, guint num_samples)                                \
  {                                                                \
   type *d = data;                                                 \
                                                                   \
   while (num_samples--) {                                         \
    largetype val = *d * filter->amplification;                    \
    do {                                                           \
     if (val > MAX_##type)                                         \
       val = MAX_##type - (val - MAX_##type);                      \
     else if (val < MIN_##type)                                    \
       val = MIN_##type + (MIN_##type - val);                      \
     else                                                          \
       break;                                                      \
    } while (1);                                                   \
    *d++ = val;                                                    \
  }                                                                \
  }                                                                \
  static void                                                      \
  gst_audio_amplify_transform_##type##_noclip (GstAudioAmplify * filter,     \
    void * data, guint num_samples)                                \
  {                                                                \
   type *d = data;                                                 \
                                                                   \
   while (num_samples--)                                           \
    *d++ *= filter->amplification;                                 \
  }
```

```
#define MAKE_FLOAT_FUNCS(type)                                        \
static void                                                           \
gst_audio_amplify_transform_##type##_clip (GstAudioAmplify * filter,  \
  void * data, guint num_samples)                                     \
{                                                                     \
 type *d = data;                                                      \
                                                                      \
 while (num_samples--) {                                              \
  type val = *d* filter->amplification;                              \
  *d++ = CLAMP (val, -1.0, +1.0);                                    \
 }                                                                    \
}                                                                     \
static void                                                           \
gst_audio_amplify_transform_##type##_wrap_negative (GstAudioAmplify * \
  filter, void * data, guint num_samples)                            \
{                                                                     \
 type *d = data;                                                      \
                                                                      \
 while (num_samples--) {                                              \
  type val = *d * filter->amplification;                            \
  do {                                                               \
   if (val > 1.0)                                                    \
    val = -1.0 + (val - 1.0);                                        \
   else if (val < -1.0)                                              \
    val = 1.0 - (1.0 - val);                                         \
   else                                                              \
    break;                                                           \
  } while (1);                                                       \
  *d++ = val;                                                        \
 }                                                                    \
}                                                                     \
static void                                                           \
gst_audio_amplify_transform_##type##_wrap_positive (GstAudioAmplify * filter, \
  void * data, guint num_samples)                                     \
{                                                                     \
 type *d = data;                                                      \
                                                                      \
 while (num_samples--) {                                              \
  type val = *d* filter->amplification;                             \
  do {                                                               \
   if (val > 1.0)                                                    \
    val = 1.0 - (val - 1.0);                                         \
   else if (val < -1.0)                                              \
    val = -1.0 + (-1.0 - val);                                       \
   else                                                              \
    break;                                                           \
  } while (1);                                                       \
```

```
      *d++ = val;                                          \
    }                                                      \
  }                                                        \
static void                                                \
gst_audio_amplify_transform_##type##_noclip (GstAudioAmplify * filter,      \
    void * data, guint num_samples)                        \
{                                                          \
  type *d = data;                                          \
                                                           \
  while (num_samples--)                                    \
    *d++ *= filter->amplification;                         \
}

/* *INDENT-OFF* */
MAKE_INT_FUNCS (gint8,gint)
MAKE_INT_FUNCS (gint16,gint)
MAKE_INT_FUNCS (gint32,gint64)
MAKE_FLOAT_FUNCS (gfloat)
MAKE_FLOAT_FUNCS (gdouble)
/* *INDENT-ON* */

/* GObject vmethod implementations */

static void
gst_audio_amplify_base_init (gpointer klass)
{
  GstElementClass *element_class = GST_ELEMENT_CLASS (klass);
  GstCaps *caps;

  gst_element_class_set_details_simple (element_class, "Audio amplifier",
      "Filter/Effect/Audio",
      "Amplifies an audio stream by a given factor",
      "Sebastian Drge <slomo@circular-chaos.org>");

  caps = gst_caps_from_string (ALLOWED_CAPS);
  gst_audio_filter_class_add_pad_templates (GST_AUDIO_FILTER_CLASS (klass),
      caps);
  gst_caps_unref (caps);
}

static void
gst_audio_amplify_class_init (GstAudioAmplifyClass * klass)
{
  GObjectClass *gobject_class;

  gobject_class = (GObjectClass *) klass;
  gobject_class->set_property = gst_audio_amplify_set_property;
  gobject_class->get_property = gst_audio_amplify_get_property;
```

```
g_object_class_install_property (gobject_class, PROP_AMPLIFICATION,
    g_param_spec_float ("amplification", "Amplification",
        "Factor of amplification", -G_MAXFLOAT, G_MAXFLOAT,
        1.0,
        G_PARAM_READWRITE | GST_PARAM_CONTROLLABLE | G_PARAM_STATIC_STRINGS));

/**
 * GstAudioAmplify:clipping-method
 *
 * Clipping method: clip mode set values higher than the maximum to the
 * maximum. The wrap-negative mode pushes those values back from the
 * opposite side, wrap-positive pushes them back from the same side.
 *
 **/
g_object_class_install_property (gobject_class, PROP_CLIPPING_METHOD,
    g_param_spec_enum ("clipping-method", "Clipping method",
        "Selects how to handle values higher than the maximum",
        GST_TYPE_AUDIO_AMPLIFY_CLIPPING_METHOD, METHOD_CLIP,
        G_PARAM_READWRITE | G_PARAM_STATIC_STRINGS));

  GST_AUDIO_FILTER_CLASS (klass)->setup =
    GST_DEBUG_FUNCPTR (gst_audio_amplify_setup);
  GST_BASE_TRANSFORM_CLASS (klass)->transform_ip =
    GST_DEBUG_FUNCPTR (gst_audio_amplify_transform_ip);
}

static void
gst_audio_amplify_init (GstAudioAmplify * filter, GstAudioAmplifyClass * klass)
{
 filter->amplification = 1.0;
 gst_audio_amplify_set_process_function (filter, METHOD_CLIP,
    GST_BUFTYPE_LINEAR, 16);
 gst_base_transform_set_in_place (GST_BASE_TRANSFORM (filter), TRUE);
 gst_base_transform_set_gap_aware (GST_BASE_TRANSFORM (filter), TRUE);
}

static GstAudioAmplifyProcessFunc
gst_audio_amplify_process_function (gint clipping, gint format, gint width)
{
 static const struct process
 {
  gint format;
  gint width;
  gint clipping;
  GstAudioAmplifyProcessFunc func;
 } process[] = {
  {
```

```
    GST_BUFTYPE_FLOAT, 32, METHOD_CLIP,
        gst_audio_amplify_transform_gfloat_clip}, {
    GST_BUFTYPE_FLOAT, 32, METHOD_WRAP_NEGATIVE,
        gst_audio_amplify_transform_gfloat_wrap_negative}, {
    GST_BUFTYPE_FLOAT, 32, METHOD_WRAP_POSITIVE,
        gst_audio_amplify_transform_gfloat_wrap_positive}, {
    GST_BUFTYPE_FLOAT, 32, METHOD_NOCLIP,
        gst_audio_amplify_transform_gfloat_noclip}, {
    GST_BUFTYPE_FLOAT, 64, METHOD_CLIP,
        gst_audio_amplify_transform_gdouble_clip}, {
    GST_BUFTYPE_FLOAT, 64, METHOD_WRAP_NEGATIVE,
        gst_audio_amplify_transform_gdouble_wrap_negative}, {
    GST_BUFTYPE_FLOAT, 64, METHOD_WRAP_POSITIVE,
        gst_audio_amplify_transform_gdouble_wrap_positive}, {
    GST_BUFTYPE_FLOAT, 64, METHOD_NOCLIP,
        gst_audio_amplify_transform_gdouble_noclip}, {
    GST_BUFTYPE_LINEAR, 8, METHOD_CLIP, gst_audio_amplify_transform_gint8_clip}, {
    GST_BUFTYPE_LINEAR, 8, METHOD_WRAP_NEGATIVE,
        gst_audio_amplify_transform_gint8_wrap_negative}, {
    GST_BUFTYPE_LINEAR, 8, METHOD_WRAP_POSITIVE,
        gst_audio_amplify_transform_gint8_wrap_positive}, {
    GST_BUFTYPE_LINEAR, 8, METHOD_NOCLIP,
        gst_audio_amplify_transform_gint8_noclip}, {
    GST_BUFTYPE_LINEAR, 16, METHOD_CLIP,
        gst_audio_amplify_transform_gint16_clip}, {
    GST_BUFTYPE_LINEAR, 16, METHOD_WRAP_NEGATIVE,
        gst_audio_amplify_transform_gint16_wrap_negative}, {
    GST_BUFTYPE_LINEAR, 16, METHOD_WRAP_POSITIVE,
        gst_audio_amplify_transform_gint16_wrap_positive}, {
    GST_BUFTYPE_LINEAR, 16, METHOD_NOCLIP,
        gst_audio_amplify_transform_gint16_noclip}, {
    GST_BUFTYPE_LINEAR, 32, METHOD_CLIP,
        gst_audio_amplify_transform_gint32_clip}, {
    GST_BUFTYPE_LINEAR, 32, METHOD_WRAP_NEGATIVE,
        gst_audio_amplify_transform_gint32_wrap_negative}, {
    GST_BUFTYPE_LINEAR, 32, METHOD_WRAP_POSITIVE,
        gst_audio_amplify_transform_gint32_wrap_positive}, {
    GST_BUFTYPE_LINEAR, 32, METHOD_NOCLIP,
        gst_audio_amplify_transform_gint32_noclip}, {
    0, 0, 0, NULL}
  };
  const struct process *p;

  for (p = process; p->func; p++)
    if (p->format == format && p->width == width && p->clipping == clipping)
      return p->func;
  return NULL;
}
```

```
static gboolean
gst_audio_amplify_set_process_function (GstAudioAmplify * filter, gint
  clipping_method, gint format, gint width)
{
 GstAudioAmplifyProcessFunc process;

 /* set processing function */

 process = gst_audio_amplify_process_function (clipping_method, format, width);
 if (!process) {
  GST_DEBUG ("wrong format");
  return FALSE;
 }

 filter->process = process;
 filter->clipping_method = clipping_method;
 filter->format = format;
 filter->width = width;

 return TRUE;
}

static void
gst_audio_amplify_set_property (GObject * object, guint prop_id,
  const GValue * value, GParamSpec * pspec)
{
 GstAudioAmplify *filter = GST_AUDIO_AMPLIFY (object);

 switch (prop_id) {
  case PROP_AMPLIFICATION:
   filter->amplification = g_value_get_float (value);
   gst_base_transform_set_passthrough (GST_BASE_TRANSFORM (filter),
     filter->amplification == 1.0);
   break;
  case PROP_CLIPPING_METHOD:
   gst_audio_amplify_set_process_function (filter, g_value_get_enum (value),
     filter->format, filter->width);
   break;
  default:
   G_OBJECT_WARN_INVALID_PROPERTY_ID (object, prop_id, pspec);
   break;
 }
}

static void
gst_audio_amplify_get_property (GObject * object, guint prop_id,
  GValue * value, GParamSpec * pspec)
```

```c
{
  GstAudioAmplify *filter = GST_AUDIO_AMPLIFY (object);

  switch (prop_id) {
    case PROP_AMPLIFICATION:
      g_value_set_float (value, filter->amplification);
      break;
    case PROP_CLIPPING_METHOD:
      g_value_set_enum (value, filter->clipping_method);
      break;
    default:
      G_OBJECT_WARN_INVALID_PROPERTY_ID (object, prop_id, pspec);
      break;
  }
}

/* GstAudioFilter vmethod implementations */
static gboolean
gst_audio_amplify_setup (GstAudioFilter * base, GstRingBufferSpec * format)
{
  GstAudioAmplify *filter = GST_AUDIO_AMPLIFY (base);

  return gst_audio_amplify_set_process_function (filter,
      filter->clipping_method, format->type, format->width);
}

/* GstBaseTransform vmethod implementations */
static GstFlowReturn
gst_audio_amplify_transform_ip (GstBaseTransform * base, GstBuffer * buf)
{
  GstAudioAmplify *filter = GST_AUDIO_AMPLIFY (base);
  guint num_samples;
  GstClockTime timestamp, stream_time;

  timestamp = GST_BUFFER_TIMESTAMP (buf);
  stream_time =
      gst_segment_to_stream_time (&base->segment, GST_FORMAT_TIME, timestamp);

  GST_DEBUG_OBJECT (filter, "sync to %" GST_TIME_FORMAT,
      GST_TIME_ARGS (timestamp));

  if (GST_CLOCK_TIME_IS_VALID (stream_time))
    gst_object_sync_values (G_OBJECT (filter), stream_time);

  num_samples =
      GST_BUFFER_SIZE (buf) / (GST_AUDIO_FILTER (filter)->format.width / 8);

  if (gst_base_transform_is_passthrough (base) ||
```

```
    G_UNLIKELY (GST_BUFFER_FLAG_IS_SET (buf, GST_BUFFER_FLAG_GAP)))
  return GST_FLOW_OK;

 filter->process (filter, GST_BUFFER_DATA (buf), num_samples);

 return GST_FLOW_OK;
}
/*
 * GStreamer
 * Copyright (C) 2007 Sebastian Drge <slomo@circular-chaos.org>
 * Copyright (C) 2006 Stefan Kost <ensonic@users.sf.net>
 *
 * 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., 59 Temple Place - Suite 330,
 * Boston, MA 02111-1307, USA.
 */

#ifndef __GST_AUDIO_AMPLIFY_H__
#define __GST_AUDIO_AMPLIFY_H__

#include <gst/gst.h>
#include <gst/base/gstbasetransform.h>
#include <gst/audio/audio.h>
#include <gst/audio/gstaudiofilter.h>

G_BEGIN_DECLS
#define GST_TYPE_AUDIO_AMPLIFY          (gst_audio_amplify_get_type())
#define GST_AUDIO_AMPLIFY(obj)
(G_TYPE_CHECK_INSTANCE_CAST((obj),GST_TYPE_AUDIO_AMPLIFY,GstAudioAmplify))
#define GST_IS_AUDIO_AMPLIFY(obj)
(G_TYPE_CHECK_INSTANCE_TYPE((obj),GST_TYPE_AUDIO_AMPLIFY))
#define GST_AUDIO_AMPLIFY_CLASS(klass)   (G_TYPE_CHECK_CLASS_CAST((klass)
,GST_TYPE_AUDIO_AMPLIFY,GstAudioAmplifyClass))
#define GST_IS_AUDIO_AMPLIFY_CLASS(klass) (G_TYPE_CHECK_CLASS_TYPE((klass)
,GST_TYPE_AUDIO_AMPLIFY))
#define GST_AUDIO_AMPLIFY_GET_CLASS(obj)  (G_TYPE_INSTANCE_GET_CLASS((obj)
,GST_TYPE_AUDIO_AMPLIFY,GstAudioAmplifyClass))
```

```
typedef struct _GstAudioAmplify GstAudioAmplify;
typedef struct _GstAudioAmplifyClass GstAudioAmplifyClass;

typedef void (*GstAudioAmplifyProcessFunc) (GstAudioAmplify *, void *, guint);

struct _GstAudioAmplify
{
 GstAudioFilter audiofilter;

 gfloat amplification;

 /* < private > */
 GstAudioAmplifyProcessFunc process;
 gint clipping_method;
 gint format;
 gint width;
};

struct _GstAudioAmplifyClass
{
 GstAudioFilterClass parent;
};

GType gst_audio_amplify_get_type (void);

G_END_DECLS
#endif /* __GST_AUDIO_AMPLIFY_H__ */
```

1.9 icu4c 56.1

1.9.1 Available under license :

ICU License - ICU 1.8.1 and later

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2012 International Business Machines Corporation and others

All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
documentation files (the "Software"), to deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to
whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice
appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in
supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,

INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR 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.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.
UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

   Unicode Data Files include all data files under the directories
http://www.unicode.org/Public/, http://www.unicode.org/reports/, and
http://www.unicode.org/cldr/data/. Unicode Data Files do not include PDF
online code charts under the directory http://www.unicode.org/Public/.
Software includes any source code published in the Unicode Standard or under
the directories http://www.unicode.org/Public/,
http://www.unicode.org/reports/, and http://www.unicode.org/cldr/data/.

   NOTICE TO USER: Carefully read the following legal agreement. BY
DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S DATA FILES
("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"), YOU UNEQUIVOCALLY ACCEPT, AND
AGREE TO BE BOUND BY, ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF
YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE THE DATA
FILES OR SOFTWARE.

   COPYRIGHT AND PERMISSION NOTICE

   Copyright © 1991-2012 Unicode, Inc. All rights reserved. Distributed under
the Terms of Use in http://www.unicode.org/copyright.html.

   Permission is hereby granted, free of charge, to any person obtaining a
copy of the Unicode data files and any associated documentation (the "Data
Files") or Unicode software and any associated documentation (the "Software")
to deal in the Data Files or Software without restriction, including without
limitation the rights to use, copy, modify, merge, publish, distribute, and/or
sell copies of the Data Files or Software, and to permit persons to whom the
Data Files or Software are furnished to do so, provided that (a) the above
copyright notice(s) and this permission notice appear with all copies of the
Data Files or Software, (b) both the above copyright notice(s) and this
permission notice appear in associated documentation, and (c) there is clear
notice in each modified Data File or in the Software as well as in the
documentation associated with the Data File(s) or Software that the data or
software has been modified.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR 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 THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

1.10 jansson 2.7

1.10.1 Available under license :

Copyright (c) 2009-2014 Petri Lehtinen <petri@digip.org>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.11 json-c 0.12

1.11.1 Available under license :

Copyright (c) 2009-2012 Eric Haszlakiewicz

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"),

to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

Copyright (c) 2004, 2005 Metaparadigm Pte Ltd

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

1.12 json-c/test-driver 0.12

1.12.1 Available under license :

```
#! /bin/sh
# test-driver - basic testsuite driver script.

scriptversion=2012-06-27.10; # UTC

# Copyright (C) 2011-2013 Free Software Foundation, Inc.
```

#
This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2, or (at your option)
any later version.
#
This program 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 General Public License for more details.
#
You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>.

As a special exception to the GNU General Public License, if you
distribute this file as part of a program that contains a
configuration script generated by Autoconf, you may include it under
the same distribution terms that you use for the rest of that program.

This file is maintained in Automake, please report
bugs to <bug-automake@gnu.org> or send patches to
<automake-patches@gnu.org>.

Make unconditional expansion of undefined variables an error. This
helps a lot in preventing typo-related bugs.
GNU GENERAL PUBLIC LICENSE
        Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

        Preamble

 The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Lesser General Public License instead.) You can apply it to
your programs, 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 software, or if you modify it.

 For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

 We protect your rights with two steps: (1) copyright the software, and
(2) offer you this license which gives you legal permission to copy,
distribute and/or modify the software.

 Also, for each author's protection and ours, we want to make certain
that everyone understands that there is no warranty for this free
software. If the software is modified by someone else and passed on, we
want its recipients to know that what they have is not the original, 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 redistributors of a free
program will individually obtain patent licenses, in effect making the
program proprietary. To prevent this, we have made it clear that any
patent must be licensed for everyone's free use or not licensed at all.

 The precise terms and conditions for copying, distribution and
modification follow.

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

 0. This License applies to any program or other work which contains
a notice placed by the copyright holder saying it may be distributed
under the terms of this General Public License. The "Program", below,
refers to any such program or work, and a "work based on the Program"
means either the Program or any derivative work under copyright law:
that is to say, a work containing the Program or a portion of it,
either verbatim or with modifications and/or translated into another
language. (Hereinafter, translation is included without limitation in
the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's 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 give any other recipients of the Program a copy of this License along with the Program.

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 Program or any portion of it, thus forming a work based on the Program, 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) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, 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 Program, 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 Program.

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

 3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of
Sections 1 and 2 above provided that you also do one of the following:

  a) 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; or,

  b) Accompany it with a written offer, valid for at least three
  years, to give any third party, for a charge no more than your
  cost of physically performing source distribution, a complete
  machine-readable copy of the corresponding source code, to be
  distributed under the terms of Sections 1 and 2 above on a medium
  customarily used for software interchange; or,

  c) Accompany it with the information you received as to the offer
  to distribute corresponding source code. (This alternative is
  allowed only for noncommercial distribution and only if you
  received the program in object code or executable form with such
  an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it. For an executable work, 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 executable. 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.

If distribution of executable or 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 counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program 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.

5. 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 Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program 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.

7. 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 Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program 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 Program.

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.

 8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program 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.

 9. The Free Software Foundation may publish revised and/or new versions
of the 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 Program
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 Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

 10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, 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

 11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "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 PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

  12. 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 PROGRAM 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 PROGRAM (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 PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), 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 Programs

 If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

 To do so, attach the following notices to the program. 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 program's name and a brief idea of what it does.>
    Copyright (C) <year> <name of author>

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

    This program 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 General Public License for more details.

    You should have received a copy of the GNU General Public License along
    with this program; 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.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than 'show w' and 'show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

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

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.

1.13 libcURL 7.65.0

1.13.1 Available under license :

License Mixing
==============

libcurl can be built to use a fair amount of various third party libraries,
libraries that are written and provided by other parties that are distributed
using their own licenses. Even libcurl itself contains code that may cause
problems to some. This document attempts to describe what licenses libcurl and
the other libraries use and what possible dilemmas linking and mixing them all
can lead to for end users.

I am not a lawyer and this is not legal advice!

One common dilemma is that [GPL](https://www.gnu.org/licenses/gpl.html)
licensed code is not allowed to be linked with code licensed under the
[Original BSD license](https://spdx.org/licenses/BSD-4-Clause.html) (with the
announcement clause). You may still build your own copies that use them all,
but distributing them as binaries would be to violate the GPL license - unless
you accompany your license with an
[exception](https://www.gnu.org/licenses/gpl-faq.html#GPLIncompatibleLibs). This
particular problem was addressed when the [Modified BSD
license](https://opensource.org/licenses/BSD-3-Clause) was created, which does
not have the announcement clause that collides with GPL.

libcurl

Uses an [MIT style license](https://curl.haxx.se/docs/copyright.html) that is
very liberal.

OpenSSL

(May be used for SSL/TLS support) Uses an Original BSD-style license with an
announcement clause that makes it "incompatible" with GPL. You are not
allowed to ship binaries that link with OpenSSL that includes GPL code
(unless that specific GPL code includes an exception for OpenSSL - a habit
that is growing more and more common). If OpenSSL's licensing is a problem
for you, consider using another TLS library.

GnuTLS

(May be used for SSL/TLS support) Uses the
[LGPL](https://www.gnu.org/licenses/lgpl.html) license. If this is a problem
for you, consider using another TLS library. Also note that GnuTLS itself
depends on and uses other libs (libgcrypt and libgpg-error) and they too are
LGPL- or GPL-licensed.

WolfSSL

(May be used for SSL/TLS support) Uses the GPL license or a proprietary
license. If this is a problem for you, consider using another TLS library.

NSS

(May be used for SSL/TLS support) Is covered by the
[MPL](https://www.mozilla.org/MPL/) license, the GPL license and the LGPL
license. You may choose to license the code under MPL terms, GPL terms, or
LGPL terms. These licenses grant you different permissions and impose
different obligations. You should select the license that best meets your
needs.

mbedTLS

(May be used for SSL/TLS support) Uses the [Apache 2.0
license](https://opensource.org/licenses/Apache-2.0) or the GPL license.
You may choose to license the code under Apache 2.0 terms or GPL terms.
These licenses grant you different permissions and impose different
obligations. You should select the license that best meets your needs.

BoringSSL

(May be used for SSL/TLS support) As an OpenSSL fork, it has the same
license as that.

libressl

(May be used for SSL/TLS support) As an OpenSSL fork, it has the same
license as that.

c-ares

(Used for asynchronous name resolves) Uses an MIT license that is very
liberal and imposes no restrictions on any other library or part you may link
with.

zlib

(Used for compressed Transfer-Encoding support) Uses an MIT-style license
that shouldn't collide with any other library.

MIT Kerberos

(May be used for GSS support) MIT licensed, that shouldn't collide with any
other parts.

Heimdal

(May be used for GSS support) Heimdal is Original BSD licensed with the
announcement clause.

GNU GSS

(May be used for GSS support) GNU GSS is GPL licensed. Note that you may not
distribute binary curl packages that uses this if you build curl to also link
and use any Original BSD licensed libraries!

libidn

(Used for IDNA support) Uses the GNU Lesser General Public License [3]. LGPL
is a variation of GPL with slightly less aggressive "copyleft". This license

requires more requirements to be met when distributing binaries, see the license for details. Also note that if you distribute a binary that includes this library, you must also include the full LGPL license text. Please properly point out what parts of the distributed package that the license addresses.

OpenLDAP

(Used for LDAP support) Uses a Modified BSD-style license. Since libcurl uses OpenLDAP as a shared library only, I have not heard of anyone that ships OpenLDAP linked with libcurl in an app.

libssh2

(Used for scp and sftp support) libssh2 uses a Modified BSD-style license. COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1996 - 2019, Daniel Stenberg, <daniel@haxx.se>, and many contributors, see the THANKS file.

All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

1.14 libjpeg 9b

1.15 liboil 0.3.16.1

1.15.1 Available under license :

The majority of the source code and the collective work is subject
to the following license:

The source code in the liboil/motovec directory is subject to the
following license:

REGARD TO THE SOFTWARE (INCLUDING ANY MODIFIED VERSIONS
THEREOF) AND ANY ACCOMPANYING WRITTEN MATERIALS.

To the maximum extent permitted by applicable law, IN NO EVENT SHALL
MOTOROLA BE LIABLE FOR ANY DAMAGES WHATSOEVER
(INCLUDING WITHOUT LIMITATION, DAMAGES FOR LOSS OF
BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS
INFORMATION, OR OTHER PECUNIARY LOSS) ARISING OF THE USE OR
INABILITY TO USE THE SOFTWARE.   Motorola assumes no responsibility
for the maintenance and support of the SOFTWARE.

The source code implementing the Mersenne Twister algorithm is
subject to the following license:

Copyright (C) 1997 - 2002, Makoto Matsumoto and Takuji Nishimura,
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

  1. Redistributions of source code must retain the above copyright
     notice, this list of conditions and the following disclaimer.

  2. Redistributions in binary form must reproduce the above copyright
     notice, this list of conditions and the following disclaimer in the
     documentation and/or other materials provided with the distribution.

  3. The names of its contributors may not be used to endorse or promote
     products derived from this software without specific prior written
     permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
/*
* LIBOIL - Library of Optimized Inner Loops
* Copyright (c) 2006 David A. Schleef <ds@schleef.org>
* All rights reserved.

```
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
* WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED.  IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT,
* INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
* (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
* SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
* STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING
* IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
* POSSIBILITY OF SUCH DAMAGE.
*/
```

1.16 libSRTP 2.2.0

1.16.1 Available under license :

```
/*
*
* Copyright (c) 2001-2017 Cisco Systems, Inc.
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
*   Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
*
*   Redistributions in binary form must reproduce the above
*   copyright notice, this list of conditions and the following
*   disclaimer in the documentation and/or other materials provided
*   with the distribution.
*
*   Neither the name of the Cisco Systems, Inc. nor the names of its
*   contributors may be used to endorse or promote products derived
*   from this software without specific prior written permission.
*
```

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS

* "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT

* LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS

* FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE

* COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,

* INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

* (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR

* SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)

* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,

* STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)

* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED

* OF THE POSSIBILITY OF SUCH DAMAGE.

*

*/

1.17 libxml2 2.9.9

1.17.1 Available under license :

Except where otherwise noted in the source code (e.g. the files hash.c,

list.c and the trio files, which are covered by a similar licence but

with different Copyright notices) all the files are:

Copyright (C) 1998-2012 Daniel Veillard. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy

of this software and associated documentation files (the "Software"), to deal

in the Software without restriction, including without limitation the rights

to use, copy, modify, merge, publish, distribute, sublicense, and/or sell

copies of the Software, and to permit persons to whom the Software is fur-

nished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in

all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR

IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FIT-

NESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE

AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER

LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,

OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN

THE SOFTWARE.

1.18 OpenSSL 1.0.2r :1.0.2r

1.18.1 Notifications :

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com).

This product includes software written by Tim Hudson (tjh@cryptsoft.com).

1.18.2 Available under license :

Copyright (C) 1995-1997 Eric Young (eay@cryptsoft.com)

All rights reserved.

This package is an Blowfish implementation written

by Eric Young (eay@cryptsoft.com).

This library is free for commercial and non-commercial use as long as

the following conditions are aheared to. The following conditions

apply to all code found in this distribution.

Copyright remains Eric Young's, and as such any Copyright notices in

the code are not to be removed.

Redistribution and use in source and binary forms, with or without

modification, are permitted provided that the following conditions

are met:

1. Redistributions of source code must retain the copyright

   notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright

   notice, this list of conditions and the following disclaimer in the

   documentation and/or other materials provided with the distribution.

3. All advertising materials mentioning features or use of this software

   must display the following acknowledgement:

   This product includes software developed by Eric Young (eay@cryptsoft.com)

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND

ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE

IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE

ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE

FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL

DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS

OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)

HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT

LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY

OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF

SUCH DAMAGE.

The license and distribution terms for any publically available version or

derivative of this code cannot be changed. i.e. this code cannot simply be

copied and put under another distrubution license

[including the GNU Public License.]

The reason behind this being stated in this direct manner is past
experience in code simply being copied and the attribution removed
from it and then being distributed as part of other packages. This
implementation was a non-trivial and unpaid effort.
Copyright (C) 1995-1997 Eric Young (eay@cryptsoft.com)
All rights reserved.

This package is an DES implementation written by Eric Young (eay@cryptsoft.com).
The implementation was written so as to conform with MIT's libdes.

This library is free for commercial and non-commercial use as long as
the following conditions are aheared to. The following conditions
apply to all code found in this distribution.

Copyright remains Eric Young's, and as such any Copyright notices in
the code are not to be removed.
If this package is used in a product, Eric Young should be given attribution
as the author of that the SSL library. This can be in the form of a textual
message at program startup or in documentation (online or textual) provided
with the package.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software
   must display the following acknowledgement:
   This product includes software developed by Eric Young (eay@cryptsoft.com)

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

The license and distribution terms for any publically available version or

derivative of this code cannot be changed. i.e. this code cannot simply be
copied and put under another distrubution license
[including the GNU Public License.]

The reason behind this being stated in this direct manner is past
experience in code simply being copied and the attribution removed
from it and then being distributed as part of other packages. This
implementation was a non-trivial and unpaid effort.

LICENSE ISSUES

==============

The OpenSSL toolkit stays under a double license, i.e. both the conditions of
the OpenSSL License and the original SSLeay license apply to the toolkit.
See below for the actual license texts. Actually both licenses are BSD-style
Open Source licenses. In case of any license issues related to OpenSSL
please contact openssl-core@openssl.org.

OpenSSL License

```
/* ====================================================================
 * Copyright (c) 1998-2018 The OpenSSL Project.  All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in
 *    the documentation and/or other materials provided with the
 *    distribution.
 *
 * 3. All advertising materials mentioning features or use of this
 *    software must display the following acknowledgment:
 *    "This product includes software developed by the OpenSSL Project
 *    for use in the OpenSSL Toolkit. (http://www.openssl.org/)"
 *
 * 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to
 *    endorse or promote products derived from this software without
 *    prior written permission. For written permission, please contact
 *    openssl-core@openssl.org.
 *
 * 5. Products derived from this software may not be called "OpenSSL"
 *    nor may "OpenSSL" appear in their names without prior written
```

```
*    permission of the OpenSSL Project.
*
* 6. Redistributions of any form whatsoever must retain the following
*    acknowledgment:
*    "This product includes software developed by the OpenSSL Project
*    for use in the OpenSSL Toolkit (http://www.openssl.org/)"
*
* THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY
* EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
* PURPOSE ARE DISCLAIMED.  IN NO EVENT SHALL THE OpenSSL PROJECT OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
* STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
* OF THE POSSIBILITY OF SUCH DAMAGE.
* ====================================================================
*
* This product includes cryptographic software written by Eric Young
* (eay@cryptsoft.com).  This product includes software written by Tim
* Hudson (tjh@cryptsoft.com).
*
*/


Original SSLeay License
-----------------------


/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
* All rights reserved.
*
* This package is an SSL implementation written
* by Eric Young (eay@cryptsoft.com).
* The implementation was written so as to conform with Netscapes SSL.
*
* This library is free for commercial and non-commercial use as long as
* the following conditions are aheared to.  The following conditions
* apply to all code found in this distribution, be it the RC4, RSA,
* lhash, DES, etc., code; not just the SSL code.  The SSL documentation
* included with this distribution is covered by the same copyright terms
* except that the holder is Tim Hudson (tjh@cryptsoft.com).
*
* Copyright remains Eric Young's, and as such any Copyright notices in
* the code are not to be removed.
* If this package is used in a product, Eric Young should be given attribution
* as the author of the parts of the library used.
```

* This can be in the form of a textual message at program startup or
* in documentation (online or textual) provided with the package.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the copyright
*   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in the
*   documentation and/or other materials provided with the distribution.
* 3. All advertising materials mentioning features or use of this software
*   must display the following acknowledgement:
*   "This product includes cryptographic software written by
*   Eric Young (eay@cryptsoft.com)"
*   The word 'cryptographic' can be left out if the rouines from the library
*   being used are not cryptographic related :-).
* 4. If you include any Windows specific code (or a derivative thereof) from
*   the apps directory (application code) you must include an acknowledgement:
*   "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
*
* THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND
* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* The licence and distribution terms for any publically available version or
* derivative of this code cannot be changed. i.e. this code cannot simply be
* copied and put under another distribution licence
* [including the GNU Public Licence.]
*/

LICENSE ISSUES
==============

The OpenSSL toolkit stays under a double license, i.e. both the conditions of
the OpenSSL License and the original SSLeay license apply to the toolkit.
See below for the actual license texts. Actually both licenses are BSD-style
Open Source licenses. In case of any license issues related to OpenSSL
please contact openssl-core@openssl.org.

OpenSSL License

```
/* ====================================================================
 * Copyright (c) 1998-2018 The OpenSSL Project.  All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in
 *    the documentation and/or other materials provided with the
 *    distribution.
 *
 * 3. All advertising materials mentioning features or use of this
 *    software must display the following acknowledgment:
 *    "This product includes software developed by the OpenSSL Project
 *    for use in the OpenSSL Toolkit. (http://www.openssl.org/)"
 *
 * 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to
 *    endorse or promote products derived from this software without
 *    prior written permission. For written permission, please contact
 *    openssl-core@openssl.org.
 *
 * 5. Products derived from this software may not be called "OpenSSL"
 *    nor may "OpenSSL" appear in their names without prior written
 *    permission of the OpenSSL Project.
 *
 * 6. Redistributions of any form whatsoever must retain the following
 *    acknowledgment:
 *    "This product includes software developed by the OpenSSL Project
 *    for use in the OpenSSL Toolkit (http://www.openssl.org/)"
 *
 * THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY
 * EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
 * PURPOSE ARE DISCLAIMED.  IN NO EVENT SHALL THE OpenSSL PROJECT OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
```

* OF THE POSSIBILITY OF SUCH DAMAGE.
* ==
*
* This product includes cryptographic software written by Eric Young
* (eay@cryptsoft.com). This product includes software written by Tim
* Hudson (tjh@cryptsoft.com).
*
*/

Original SSLeay License

/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
* All rights reserved.
*
* This package is an SSL implementation written
* by Eric Young (eay@cryptsoft.com).
* The implementation was written so as to conform with Netscapes SSL.
*
* This library is free for commercial and non-commercial use as long as
* the following conditions are aheared to. The following conditions
* apply to all code found in this distribution, be it the RC4, RSA,
* lhash, DES, etc., code; not just the SSL code. The SSL documentation
* included with this distribution is covered by the same copyright terms
* except that the holder is Tim Hudson (tjh@cryptsoft.com).
*
* Copyright remains Eric Young's, and as such any Copyright notices in
* the code are not to be removed.
* If this package is used in a product, Eric Young should be given attribution
* as the author of the parts of the library used.
* This can be in the form of a textual message at program startup or
* in documentation (online or textual) provided with the package.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the copyright
*   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in the
*   documentation and/or other materials provided with the distribution.
* 3. All advertising materials mentioning features or use of this software
*   must display the following acknowledgement:
*   "This product includes cryptographic software written by
*   Eric Young (eay@cryptsoft.com)"
*   The word 'cryptographic' can be left out if the rouines from the library
*   being used are not cryptographic related :-).
* 4. If you include any Windows specific code (or a derivative thereof) from

*   the apps directory (application code) you must include an acknowledgement:

*   "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

*

* THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND

* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE

* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE

* ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE

* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL

* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS

* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)

* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT

* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY

* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF

* SUCH DAMAGE.

*

* The licence and distribution terms for any publically available version or

* derivative of this code cannot be changed. i.e. this code cannot simply be

* copied and put under another distribution licence

* [including the GNU Public Licence.]

*/

1.19 opus 1.1

1.19.1 Available under license :

Copyright 2001-2011 Xiph.Org, Skype Limited, Octasic,
            Jean-Marc Valin, Timothy B. Terriberry,
            CSIRO, Gregory Maxwell, Mark Borgerding,
            Erik de Castro Lopo

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

- Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

- Neither the name of Internet Society, IETF or IETF Trust, nor the
names of specific contributors, may be used to endorse or promote
products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS

``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Opus is subject to the royalty-free patent licenses which are specified at:

Xiph.Org Foundation:
https://datatracker.ietf.org/ipr/1524/

Microsoft Corporation:
https://datatracker.ietf.org/ipr/1914/

Broadcom Corporation:
https://datatracker.ietf.org/ipr/1526/

1.20 protobuf-cpp 3.7.1

1.20.1 Available under license :

Copyright 2008 Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

   * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
   * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
   * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,

SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.
This file contains a list of people who have made large contributions to the public version of Protocol Buffers.

Original Protocol Buffers design and implementation:
 Sanjay Ghemawat <sanjay@google.com>
 Jeff Dean <jeff@google.com>
 Daniel Dulitz <daniel@google.com>
 Craig Silverstein
 Paul Haahr <haahr@google.com>
 Corey Anderson <corin@google.com>
 (and many others)

Proto2 C++ and Java primary author:
 Kenton Varda <kenton@google.com>

Proto2 Python primary authors:
 Will Robinson <robinson@google.com>
 Petar Petrov <petar@google.com>

Java Nano primary authors:
 Brian Duff <bduff@google.com>
 Tom Chao <chaot@google.com>
 Max Cai <maxtroy@google.com>
 Ulas Kirazci <ulas@google.com>

Large code contributions:
 Jason Hsueh <jasonh@google.com>
 Joseph Schorr <jschorr@google.com>
 Wenbo Zhu <wenboz@google.com>

Large quantity of code reviews:
 Scott Bruce <sbruce@google.com>
 Frank Yellin
 Neal Norwitz <nnorwitz@google.com>
 Jeffrey Yasskin <jyasskin@google.com>
 Ambrose Feinstein <ambrose@google.com>

Documentation:
 Lisa Carey <lcarey@google.com>

Maven packaging:
 Gregory Kick <gak@google.com>

Patch contributors:
 Kevin Ko <kevin.s.ko@gmail.com>
   * Small patch to handle trailing slashes in --proto_path flag.
 Johan Euphrosine <proppy@aminche.com>
   * Small patch to fix Python CallMethod().
 Ulrich Kunitz <kune@deine-taler.de>
   * Small optimizations to Python serialization.
 Leandro Lucarella <llucax@gmail.com>
   * VI syntax highlighting tweaks.
   * Fix compiler to not make output executable.
 Dilip Joseph <dilip.antony.joseph@gmail.com>
   * Heuristic detection of sub-messages when printing unknown fields in
     text format.
 Brian Atkinson <nairb774@gmail.com>
   * Added @Override annotation to generated Java code where appropriate.
 Vincent Choinire <Choiniere.Vincent@hydro.qc.ca>
   * Tru64 support.
 Monty Taylor <monty.taylor@gmail.com>
   * Solaris 10 + Sun Studio fixes.
 Alek Storm <alek.storm@gmail.com>
   * Slicing support for repeated scalar fields for the Python API.
 Oleg Smolsky <oleg.smolsky@gmail.com>
   * MS Visual Studio error format option.
   * Detect unordered_map in stl_hash.m4.
 Brian Olson <brianolson@google.com>
   * gzip/zlib I/O support.
 Michael Poole <mdpoole@troilus.org>
   * Fixed warnings about generated constructors not explicitly initializing
     all fields (only present with certain compiler settings).
   * Added generation of field number constants.
 Wink Saville <wink@google.com>
   * Fixed initialization ordering problem in logging code.
 Will Pierce <willp@nuclei.com>
   * Small patch improving performance of in Python serialization.
 Alexandre Vassalotti <alexandre@peadrop.com>
   * Emacs mode for Protocol Buffers (editors/protobuf-mode.el).
 Scott Stafford <scott.stafford@gmail.com>
   * Added Swap(), SwapElements(), and RemoveLast() to Reflection interface.
 Alexander Melnikov <alm@sibmail.ru>
   * HPUX support.
 Oliver Jowett <oliver.jowett@gmail.com>
   * Detect whether zlib is new enough in configure script.

    * Fixes for Solaris 10 32/64-bit confusion.

Evan Jones <evanj@mit.edu>

    * Optimize Java serialization code when writing a small message to a stream.

    * Optimize Java serialization of strings so that UTF-8 encoding happens only

      once per string per serialization call.

    * Clean up some Java warnings.

    * Fix bug with permanent callbacks that delete themselves when run.

Michael Kucharski <m.kucharski@gmail.com>

    * Added CodedInputStream.getTotalBytesRead().

Kacper Kowalik <xarthisius.kk@gmail.com>

    * Fixed m4/acx_pthread.m4 problem for some Linux distributions.

William Orr <will@worrbase.com>

    * Fixed detection of sched_yield on Solaris.

    * Added atomicops for Solaris

Andrew Paprocki <andrew@ishiboo.com>

    * Fixed minor IBM xlC compiler build issues

    * Added atomicops for AIX (POWER)

Apache License

Version 2.0, January 2004

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally

submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of
   this License, each Contributor hereby grants to You a perpetual,
   worldwide, non-exclusive, no-charge, royalty-free, irrevocable
   copyright license to reproduce, prepare Derivative Works of,
   publicly display, publicly perform, sublicense, and distribute the
   Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of
   this License, each Contributor 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, offer to sell, sell, import, and otherwise transfer the Work,
   where such license applies only to those patent claims licensable
   by such Contributor that are necessarily infringed by their
   Contribution(s) alone or by combination of their Contribution(s)
   with the Work to which such Contribution(s) was submitted. If You
   institute patent litigation against any entity (including a
   cross-claim or counterclaim in a lawsuit) alleging that the Work
   or a Contribution incorporated within the Work constitutes direct
   or contributory patent infringement, then any patent licenses
   granted to You under this License for that Work shall terminate
   as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the
   Work or Derivative Works thereof in any medium, with or without
   modifications, and in Source or Object form, provided that You
   meet the following conditions:

   (a) You must give any other recipients of the Work or
       Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices
       stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
    that You distribute, all copyright, patent, trademark, and
    attribution notices from the Source form of the Work,
    excluding those notices that do not pertain to any part of
    the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
    distribution, then any Derivative Works that You distribute must
    include a readable copy of the attribution notices contained
    within such NOTICE file, excluding those notices that do not
    pertain to any part of the Derivative Works, in at least one
    of the following places: within a NOTICE text file distributed
    as part of the Derivative Works; within the Source form or
    documentation, if provided along with the Derivative Works; or,
    within a display generated by the Derivative Works, if and
    wherever such third-party notices normally appear. The contents
    of the NOTICE file are for informational purposes only and
    do not modify the License. You may add Your own attribution
    notices within Derivative Works that You distribute, alongside
    or as an addendum to the NOTICE text from the Work, provided
    that such additional attribution notices cannot be construed
    as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
   any Contribution intentionally submitted for inclusion in the Work
   by You to the Licensor shall be under the terms and conditions of
   this License, without any additional terms or conditions.
   Notwithstanding the above, nothing herein shall supersede or modify
   the terms of any separate license agreement you may have executed
   with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
   names, trademarks, service marks, or product names of the Licensor,
   except as required for reasonable and customary use in describing the
   origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
   agreed to in writing, Licensor provides the Work (and each
   Contributor provides its Contributions) on an "AS IS" BASIS,
   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or

implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
   whether in tort (including negligence), contract, or otherwise,
   unless required by applicable law (such as deliberate and grossly
   negligent acts) or agreed to in writing, shall any Contributor be
   liable to You for damages, including any direct, indirect, special,
   incidental, or consequential damages of any character arising as a
   result of this License or out of the use or inability to use the
   Work (including but not limited to damages for loss of goodwill,
   work stoppage, computer failure or malfunction, or any and all
   other commercial damages or losses), even if such Contributor
   has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
   the Work or Derivative Works thereof, You may choose to offer,
   and charge a fee for, acceptance of support, warranty, indemnity,
   or other liability obligations and/or rights consistent with this
   License. However, in accepting such obligations, You may act only
   on Your own behalf and on Your sole responsibility, not on behalf
   of any other Contributor, and only if You agree to indemnify,
   defend, and hold each Contributor harmless for any liability
   incurred by, or claims asserted against, such Contributor by reason
   of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

   To apply the Apache License to your work, attach the following
   boilerplate notice, with the fields enclosed by brackets "[]"
   replaced with your own identifying information. (Don't include
   the brackets!) The text should be enclosed in the appropriate
   comment syntax for the file format. We also recommend that a
   file or class name and description of purpose be included on the
   same "printed page" as the copyright notice for easier
   identification within third-party archives.

Copyright [2007] Neal Norwitz
Portions Copyright [2007] Google Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
This file contains a list of people who've made non-trivial
contribution to the Google C++ Mocking Framework project. People
who commit code to the project are encouraged to add their names
here. Please keep the list sorted by first names.

Benoit Sigoure <tsuna@google.com>
Bogdan Piloca <boo@google.com>
Chandler Carruth <chandlerc@google.com>
Dave MacLachlan <dmaclach@gmail.com>
David Anderson <danderson@google.com>
Dean Sturtevant
Gene Volovich <gv@cite.com>
Hal Burch <gmock@hburch.com>
Jeffrey Yasskin <jyasskin@google.com>
Jim Keller <jimkeller@google.com>
Joe Walnes <joe@truemesh.com>
Jon Wray <jwray@google.com>
Keir Mierle <mierle@gmail.com>
Keith Ray <keith.ray@gmail.com>
Kostya Serebryany <kcc@google.com>
Lev Makhlis
Manuel Klimek <klimek@google.com>
Mario Tanev <radix@google.com>
Mark Paskin
Markus Heule <markus.heule@gmail.com>
Matthew Simmons <simmonmt@acm.org>
Mike Bland <mbland@google.com>
Neal Norwitz <nnorwitz@gmail.com>
Nermin Ozkiranartli <nermin@google.com>
Owen Carlsen <ocarlsen@google.com>
Paneendra Ba <paneendra@google.com>
Paul Menage <menage@google.com>
Piotr Kaminski <piotrk@google.com>
Russ Rufer <russ@pentad.com>
Sverre Sundsdal <sundsdal@gmail.com>
Takeshi Yoshino <tyoshino@google.com>
Vadim Berman <vadimb@google.com>
Vlad Losev <vladl@google.com>
Wolfgang Klier <wklier@google.com>
Zhanyong Wan <wan@google.com>

This file contains a list of people who've made non-trivial
contribution to the Google C++ Testing Framework project. People
who commit code to the project are encouraged to add their names
here. Please keep the list sorted by first names.

Ajay Joshi <jaj@google.com>
Balzs Dn <balazs.dan@gmail.com>
Bharat Mediratta <bharat@menalto.com>
Chandler Carruth <chandlerc@google.com>
Chris Prince <cprince@google.com>
Chris Taylor <taylorc@google.com>
Dan Egnor <egnor@google.com>
Eric Roman <eroman@chromium.org>
Hady Zalek <hady.zalek@gmail.com>
Jeffrey Yasskin <jyasskin@google.com>
Ji Sigursson <joi@google.com>
Keir Mierle <mierle@gmail.com>
Keith Ray <keith.ray@gmail.com>
Kenton Varda <kenton@google.com>
Manuel Klimek <klimek@google.com>
Markus Heule <markus.heule@gmail.com>
Mika Raento <mikie@iki.fi>
Mikls Fazekas <mfazekas@szemafor.com>
Pasi Valminen <pasi.valminen@gmail.com>
Patrick Hanna <phanna@google.com>
Patrick Riley <pfr@google.com>
Peter Kaminski <piotrk@google.com>
Preston Jackson <preston.a.jackson@gmail.com>
Rainer Klaffenboeck <rainer.klaffenboeck@dynatrace.com>
Russ Cox <rsc@google.com>
Russ Rufer <russ@pentad.com>
Sean Mcafee <eefacm@gmail.com>
Sigurur sgeirsson <siggi@google.com>
Tracy Bialik <tracy@pentad.com>
Vadim Berman <vadimb@google.com>
Vlad Losev <vladl@google.com>
Zhanyong Wan <wan@google.com>

distribution.
 * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner
of the input file used when generating it. This code is not
standalone and requires a support library to be linked with it. This
support library is itself covered by the above license.

1.21 sqlcipher 3.4

1.21.1 Available under license :

The author disclaims copyright to this source code. In place of
a legal notice, here is a blessing:

 May you do good and not evil.
 May you find forgiveness for yourself and forgive others.
 May you share freely, never taking more than you give.
Copyright (c) 2008, ZETETIC LLC
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
 * Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
 * Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
 * Neither the name of the ZETETIC LLC nor the
   names of its contributors may be used to endorse or promote products
   derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY ZETETIC LLC "AS IS" AND ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED

WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ZETETIC LLC BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.22 sqlite 3.28.0

1.22.1 Available under license :

The author disclaims copyright to this source code. In place of
a legal notice, here is a blessing:

 May you do good and not evil.
 May you find forgiveness for yourself and forgive others.
 May you share freely, never taking more than you give.

1.23 Zlib 1.2.3

1.23.1 Available under license :

Copyright (C) 1995-2005 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied
warranty. In no event will the authors be held liable for any damages
arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,
including commercial applications, and to alter it and redistribute it
freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not
   claim that you wrote the original software. If you use this software
   in a product, an acknowledgment in the product documentation would be
   appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be
   misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

 Jean-loup Gailly   Mark Adler
 jloup@gzip.org   madler@alumni.caltech.edu

1.24 zlib 1.2.8

1.24.1 Available under license :

Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization
obtaining a copy of the software and accompanying documentation covered by
this license (the "Software") to use, reproduce, display, distribute,
execute, and transmit the Software, and to prepare derivative works of the
Software, and to permit third-parties to whom the Software is furnished to
do so, all subject to the following:

The copyright notices in the Software and this entire statement, including
the above license grant, this restriction and the following disclaimer,
must be included in all copies of the Software, in whole or in part, and
all derivative works of the Software, unless such copies or derivative
works are solely in the form of machine-executable object code generated by
a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT
SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE
FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.