

Release Notes

Cygnus Support Developer's Kit
Progressive-96q1

Cygnus Support

Edited by Victoria Mixon (doc@cygnus.com).

Copyright © 1991, 1992, 1993, 1994, 1995, 1996 Cygnus Support

Permission is granted to make and distribute verbatim copies of this manual provided the copyright notice and this permission notice are preserved on all copies.

Permission is granted to copy and distribute modified versions of this manual under the conditions for verbatim copying, provided also that the entire resulting derived work is distributed under the terms of a permission notice identical to this one.

Permission is granted to copy and distribute translations of this manual into another language, under the above conditions for modified versions.

Table of Contents

Cygnus Developer's Kit: Progressive-96q1	1
Using <code>send-pr</code> for questions and bug reports	3
Checking the Cygnus bug database by mail	3
Confidential information in problem reports	4
New in this release	5
HPUX support	5
H8/300 <code>#pragma</code> section support obsolescent	5
Shipping <code>GAS</code> , <code>LD</code> , and <code>binutils</code> on <code>AIX</code>	5
What's new with <code>G++</code>	5
What's new with <code>GDB</code>	6
What's new with <code>GAS</code>	7
What's new with <code>LD</code>	8
Limitations and Warnings	9
Programs not available on some platforms	9
All <code>QIC-24</code> platforms now shipping on two tapes	9
Linking with <code>libg++</code>	9
<code>GDB</code> connections problem	10
<code>expect</code> generates bad results in an Emacs buffer	10
Issues from previous releases	11
<code>-gstabs</code> gives better <code>C++</code> <code>COFF</code> debugging	11
Options for <code>CPU32</code> and <code>CPU32+</code> targets	11
Debugging remote connections	11
Requirements for <code>MS-DOS</code>	12
<code>DEL</code> does not work in <code>MS-DOS</code> Info	12
Notes on rebuilding from source	13
Need to upgrade <code>XLC</code> to rebuild on <code>AIX 3.2X</code>	13
Use <code>'--with-gnu-as'</code> when configuring <code>MIPS</code>	13
Rebuilding the tools under <code>Solaris 2</code>	14
Multiple object code formats	14
Problems fixed in this release	15
Binary Utilities	15
Problems rebuilding from source	15
GNU <code>C++</code> Compiler (<code>G++</code>)	15
GNU Assembler (<code>AS</code>)	17
GNU <code>C</code> Compiler (<code>GCC</code>)	17

GNU Debugger (GDB)	18
Miscellaneous GNU bug issues (gnats)	18
Problems installing the CDK	18
GNU Linker (LD)	18
Cygnus C Support Library (libc)	19
Both Cygnus C Support Libraries (libc and libm)	19
Cygnus Support make	19
Cygnus Support Problem Report Management System (PRMS)	19
Cygnus Support Problem Report Submission (send-pr)	19

**Appendix A Specifying Names for Hosts and
Targets 21**

Host names	21
config.guess	21

**Appendix B Graphical User Interface for GDB
..... 23**

Getting Started	23
Menus	24
Windows	25
Rebuilding the GDB GUI	26

Cygnus Developer's Kit: Progressive-96q1

The progressive-96q1 release is available for these native and cross-development configurations:

Host	OS	Target	Output	Monitor	New
DEC Alpha	Digital Unix 2.0	Native			
DEC Alpha	Digital Unix 3.2C	Native			
HP 9000/300	HPUX 9.00	Native			
		68K	a.out		
HP 9000/700	HPUX 9.05	68K	a.out		
		68K	COFF		
		68K-VxWorks 5.1			
		Hitachi H8/300	COFF		
		i960-VxWorks 5.1			
		Native			
		PowerPC EABI	ELF		**
HP 9000/700	HPUX 10.01	Native			
i386	DOS	68K	a.out		
		Hitachi H8/300	COFF		
		Hitachi SH	COFF		
		i386	a.out		
		IDT R3000	ELF		
		IDT R3000	ECOFF		
		IDT R4xx0	ELF		
		PowerPC EABI	ELF		**
		SPARC	a.out		
		SPARCelite	a.out		
		SPARCelite	COFF		
IBM PowerPC	AIX 4.1.3	Native			
		PowerPC EABI	ELF		**
IBM RS/6000	AIX 3.2.5	Native			
		68K	a.out		
		i960-VxWorks 5.1			
		PowerPC EABI	ELF		
IBM RS/6000	AIX 4.1.4	Native			
SGI Iris	IRIX 5.3	Native			
		Hitachi SH	COFF		
		IDT R3000	ELF		
		IDT R3000	ECOFF		
		IDT R4xx0	ELF		
SPARC	Solaris 2.4	Native			
		AMD 29K	COFF	UDI	
		68K	a.out		
		68K	COFF		
		68K-VxWorks 5.1			
		i960-VxWorks 5.1			
		PowerPC EABI	ELF		**
		SPARCelite	a.out		
		SPARCelite	COFF		

(New products are indicated with **)

Host	OS	Target	Output	Monitor	New
SPARC	SunOS 4.1.4	Native			
		AMD 29K	COFF	UDI	
		68K	a.out		
		68K	COFF		
		68K	ELF		
		68K-VxWorks 5.1			
		AMD 29K VxWorks 5.1			
		Hitachi H8/300	COFF		
		Hitachi SH	COFF		
		i386	a.out		
		i386	ELF		
		i960	COFF	Nindy	
		i960-VxWorks 5.1			
		PowerPC EABI	ELF		
		SPARC	a.out		
		SPARCelite	a.out		
		SPARCelite	COFF		
		IDT R3000	ELF		
		IDT R3000	ECOFF		
		IDT R4xx0	ELF		

(New products are indicated with **)

Futures

- **Solaris**
We will move to Solaris 2.5 in 96Q2.
- **Linux**
We will offer Linux native tools running on a 486 or better processor in 96Q2; this will be distributed on CDROM.

For discussion of the three-part naming scheme used to configure your software for each host/target combination, see Appendix A “Specifying Names for Hosts and Targets”, page 21.

These are the current version numbers for the individual programs in the progressive-96q1 release:

<i>Program</i>	<i>Cygnus Version Numbers</i>
bfd	2.6-96q1
binutils	2.6-96q1
byacc	+28-96q1
diff	2.7-96q1
dosrel	2.2-96q1
expect	5.18.1
flex	2.5.2-96q1
gas	2.6-96q1
gasp	1.2-96q1
gcc	2.7-96q1
gcov	1.5-96q1
gdb	4.15-96q1
ld	2.6-96q1
libg++	2.7-96q1
libio	2.7-96q1
libg++	2.7-96q1
libio	2.7-96q1
make	3.74-96q1
makeinfo	1.55-96q1
newlib/libc	1.6-96q1
newlib/libm	1.6-96q1
patch	2.1-96q1
send-pr	3.97-96q1
texindex	1.45-96q1
texinfo.tex	2.122-96q1

Using send-pr for questions and bug reports

Use `send-pr` for both questions and bug reports. `send-pr` ensures that questions and bug reports are tracked and routed directly to the appropriate person.

Checking the Cygnus bug database by mail

You can interrogate non-confidential bug reports in the Cygnus Problem Report Management System (PRMS) by electronic mail.

Send mail to `'query-pr@cygnus.com'`, with query parameters in the `'Subject:'` line of your mail header. (The message body is ignored.)

For example, to inquire about problem reports numbered 4020 and 5004, send mail including these lines in the header:

```
To: query-pr@cygnus.com
Subject: 4020 5004
```

You can also include many command line options to request information on bugs in a particular state, or a particular category; for example, this header requests a list of all open G++ bugs that are not confidential:

```
To: query-pr@cygnus.com
Subject: --state=open --category=g++
```

If you do not include a '`--state=`' specification in your subject line, the mail server uses

```
--state="open|analyzed|feedback|suspended"
```

Careful! Since the default state specification for electronic mail queries does not include `closed`, no news is good news—a closed bug yields a response with no message body.

Also, confidential bug reports are not available via the mail query server. You can request the status of your confidential PRs from your Cygnus technical contact.

Many options are available. To see a synopsis, send a message like this:

```
To: query-pr@cygnus.com
Subject: --help
```

Confidential information in problem reports

There has been some confusion about where to put confidential information in problem reports sent with `send-pr`. If you submit a problem report to Cygnus, and you want its detailed contents to remain confidential, set the '`>Confidential:`' field to '`yes`'.

However, the '`Subject:`' line in the mail header and the '`>Synopsis:`' field in the body of the PR are *not* treated as confidential information, as they are used when we compile reports, such as the list of Fixed Problems in this manual (see "Problems fixed in this release", page 15). *Do not* put confidential information in these fields. Any code samples, machine descriptions, problem details, and so on of course remain strictly confidential in any problem report marked as such.

The mail query server for problem reports never reports any information from confidential bug reports.

New in this release

There are improvements in each of the major development tools.

We've also fixed many problems. See "Problems fixed in this release", page 15, for a list of bugfixes.

HPUX support

We now provide tools for the HP 9000/700 running both versions 9.05 and 10.01. We have made an effort to try to determine which version each of our HPUX customers requires. However, if you receive the wrong version, let us know by sending email to production@cygnus.com and we'll make sure you get the correct version.

H8/300 #pragma section support obsolescent

The Hitachi h8300-hms target currently supports `#pragma section foo`, which can be used to put objects in section 'foo'. This feature will no longer be supported in `progressive-96q2`. Currently, if you use `#pragma section` you get a warning that the feature is deprecated and that section attributes should be used instead.

Shipping GAS, LD, and binutils on AIX

We now ship the GNU assembler, linker and binutils in AIX versions 3.2.5 and 4.1.4.

What's new with C++

A public review copy of the April *Draft of the ANSI C++ Standard* is now available. For PostScript and PDF (Adobe Acrobat) versions, see the archive at '<ftp://research.att.com/dist/c++std/WP>'. For HTML and ASCII versions, see '<ftp://ftp.cygnus.com/pub/g++>'. On the Web, see '<http://www.cygnus.com/misc/wp>'.

- **Exception handling**
Exception handling support has been significantly improved, though optimization is still not supported.
- **Nested types**
Definition of nested types outside of their containing class is now supported. For example:

```
struct A {
    struct B;
    B* bp;
};

struct A::B {
    int member;
};
```

- **Template constructors and destructors**

Explicit instantiation of template constructors and destructors is now supported. For example:

```
template A<int>::A(const A&);
```

- **HPPA**

On the HPPA, some classes that do not define a copy constructor will be passed and returned in memory again so that functions returning those types can be inlined.

What's new with GDB

GDB has had many bug fixes, particularly in the support of various remote targets.

For the complete matrix of supported hosts vs. targets, see "Introduction", page 1. Contact us at +1-800 CYGNUS-1 if you have any questions.

- **HPUX 10.0**

HPUX 10.0 is now supported.

- **Apple Macintosh support**

GDB now includes support for the Apple Macintosh, as a host only. GDB can be run as either an MPW tool or as a standalone application, and it can debug through the serial port. All of the usual GDB commands are available, but you must supply 'serial' as the device type to the target command, instead of '/dev/ttyXX'. For example:

```
target serial
```

See 'mpw-README' in the main directory for more information on how to build. The MPW configuration scripts '* /mpw-config.in' support only a few targets, and only the mips-idt-ecoff target has been completely tested. Both m68k and PowerPC Macs are supported.

- **target ppctest**

Use target ppctest for support for the PowerPC PPCBUG monitor.

- **PowerPC configuration simulator**

PowerPC configuration now includes a simulator contributed by Andrew Cagney.

- `target sh3`
Use `target sh3` for support for the Hitachi SH3 monitor ROM.
- `auto-solib-add=value`
Use the `auto-solib-add` variable to read in symbols from all shared libraries.
If the value of `auto-solib-add` is 1, then symbols from all shared libraries will be read in when the program starts up. This is convenient if you want to reference a symbol in a shared library without having to stop in that library first, such as in setting a breakpoint. The default value is 0, which improves startup time.
Note that the command `sharedlibrary` is always available to load shared library symbols manually.
- `dont-repeat`
Use the command `dont-repeat` in user-defined commands to defeat the auto-repeat of GDB when an empty command is entered.
- AIX symbol reader
The symbol reader for AIX GDB now uses partial symbol tables. This can greatly improve startup time, especially for large executables.
- Printing typedef variables
When printing the type of a variable declared with a typedef, GDB uses the typedef name if possible instead of the typedef definition.
- MIPS targets
Performance is improved in MIPS IDT debugging (MIPS targets), both for stepping and for downloads.
- `remotedelay`
The `remotedelay` option is set by default to 1. Loading executables can be considerably slower with `remotedelay` set to 1, but it gets around a loading bug on certain H8/300 boards.
To see a noticeable speed-up in loading when you're not using an H8/300 board, set `remotedelay` to 0. For example:

```
set remotedelay 0
```

What's new with GAS

- Macros
GAS now supports macros without requiring GASP.
- `-M`
- `--mri`
GAS now supports the `-M` or `--mri` option, permitting the assembly of MRI-format assembler files.

- **SunOS assembler**
The SunOS assembler is now able to assemble PIC code.

What's new with LD

- **-rpath-link**
The `-rpath-link` option has been added for SunOS and ELF systems.
- **COFF linker**
The COFF linker now automatically combines `struct`, `union`, and `enum` debugging information, so that the information only appears once in the output file. This only applies when using COFF debugging information, as opposed to `stabs`.
- **SunOS linker**
The SunOS linker is now able to create shared libraries. It should no longer be necessary to use the native SunOS tools for anything.

Limitations and Warnings

Programs not available on some platforms

- DEC Alpha running OSF/1 2.0
The `progressive-96q1` release does *not* include a linker (`ld`) for the Alpha. The native linker is used.
- SGI Irix
The Developer's Kit requires the operating system vendor's C library and include files in a native configuration. The SGI Irix operating system does not contain these files by default, but they are included in a separate developer's package. You cannot use the Cygnus Developer's Kit without this package.
- HP9000/700 native
GNU `ld` is not included for the HP700 in the native configuration.

All QIC-24 platforms now shipping on two tapes

We now ship all of the platforms that are on QIC-24 on two tapes, with the source on one tape and the binaries on the other. This does not affect products shipped on 8mm Exabyte tapes or 4mm DAT tapes.

To install source code from QIC-24 tapes, use:

```
./Install binaries extract fix test -installdir=. . .
```

We encourage customers to switch to 4mm DAT (preferred) or 8mm Exabyte. To update your release media or to get help solving any problems with installation, please contact us at +1 800 CYGNUS-1.

Linking with `libg++`

There are two issues when linking with `libg++`:

1. Symbols referenced by the shared library that wouldn't be referenced linking with the static library. Specifically, the `curses` functions. This problem will be fixed in the `progressive-96q2` release.
2. Finding the shared library at run time.
Finding a shared library at run-time is a problem with the (default) C++ library `libstdc++` as well as with `libg++`.

The fixes for the first issue are:

1. Don't link with `libg++` (if using `g++`, use `c++` instead).
- OR

2. Link with `-lcurses` (and possibly `-ltermcap`).
OR
3. Remove or rename completely the shared libraries under the installation directory (`/usr/progressive/lib/libg++.s*` and `/usr/progressive/lib/libstdc++.s*`).
OR
4. Link with `-static` to avoid using any shared libraries.

The fixes for the second are:

1. Add `/usr/progressive/lib` to your `LD_LIBRARY_PATH` environment variable.
OR
2. Use the appropriate option when linking:

```
... -R/usr/progressive/lib ... (for Solaris)
... -Wl,-rpath,/usr/progressive/lib ...
                        (for DEC Unix/OSF1 and Irix 5)
```

OR

3. Use options 3 or 4 above.

Note that the library directory, `/usr/progressive/lib`, may be different if you have installed your `CDK` distribution in a non-standard location. This directory can also be thought of as `$GCC_EXEC_PREFIX/./` if you are using `'GCC_EXEC_PREFIX'`.

GDB connections problem

GDB is unable to connect either the `a29k-amd-udi-gdb` configuration or the `i960-nindy-coff-gdb` configuration. When it tries to connect, it returns an error referring to magic numbers.

There's a one-line patch to correct this, available from Cygnus Support.

expect generates bad results in an Emacs buffer

If you run `DEJAGNU` in an Emacs shell buffer, `expect` will generate incorrect results for `pass-fail`.

Issues from previous releases

These issues are not necessarily bugs, but they may help provide advice if you run into problems with our release.

-gstabs gives better C++ COFF debugging

If you use the GCC option `'-gstabs'`, GCC embeds extended debugging information in COFF object files. (The extended debug information is based on the *stabs* debugging format, which was originally used only with the `a.out` object file format; see *The stabs debug format*, in your sources as `'src/gdb/doc/stabs.texinfo'`, or contact Cygnus for more information.) With this additional debugging information, you can debug C++ programs with GDB, even on systems that use COFF.

You can get better C++ debugging by compiling with `'-gstabs'` for these targets:

<code>a29k-amd-udi</code>	<code>m68k-coff</code>
<code>h8300-hms</code>	<code>m88k-coff</code>
<code>z8k-coff</code>	<code>sh-hms</code>

Options for CPU32 and CPU32+ targets

The Motorola CPU32 and CPU32+ targets are part of the family of 68000 chips which Cygnus supports. There are a few options to help you compile code for these targets:

- GCC has an option `'-m68332'` to be used specifically when compiling for the Motorola 68332 board. (GCC also has an option `'-m68302'`, currently undocumented. The 68302 technically isn't a CPU32 chip.)
- It is also possible to configure GCC for a target of `'m68332-aout'` or `'m68332-coff'` when rebuilding from source, in which case `'-m68332'` is the default.
- GNU AS accepts the following board-specific options:

<code>-mcpu32</code>	<code>-m68331</code>	<code>-m68332</code>
<code>-m68333</code>	<code>-m68340</code>	<i>(and -m68302)</i>

Contact Cygnus Support for more information on our support for CPU32 and CPU32+ targets.

Debugging remote connections

A common hurdle in cross development is to get the communications set up properly between the target board and the development platform.

The GDB 'set remotedebug' command can help. It was designed to help develop new remote targets; it displays the packets transmitted back and forth between GDB and the target environment. This command can be helpful in diagnosing communications problems, for example allowing you to observe packets not getting through, or noise on the line.

The set remotedebug command is now consistent among the MIPS remote target; remote targets using the GDB-specific protocol; UDI (the AMD debug protocol for the 29k); the 88k BUG monitor; and Hitachi ROM monitors. You can set it to an integer specifying a protocol-debug level (normally 0 or 1, but 2 means more protocol information for the MIPS target). See section "Communication protocol" in *Debugging with GDB*, for details.

Requirements for MS-DOS

The Cygnus Developer's Kit is only supported on MS-DOS 6.2 or higher.

We do not recommend using the cross-development kit with less than four (4) megabytes of RAM.

We provide a MS-DOS extender with the cross-development kit for MS-DOS which does swapping to disk when MS-DOS runs out of memory. To avoid excessive swapping you must have at least two (2) megabytes of RAM to run G++ on a PC with MS-DOS. If you've got more than two megabytes, the extra memory can be used as a disk cache to significantly improve performance.

DEL does not work in MS-DOS Info

GNU Info, the online documentation browser, is available in our MS-DOS distribution.

Unfortunately, the DOS version of Info, INFO.EXE, does not recognize the DEL key. This key is normally used for paging backwards within a node in Info.

As a workaround, you can page backwards by keying ESC v.

Notes on rebuilding from source

Details on rebuilding specific platforms and features are shown below. See the manual *Rebuilding From Source* for detailed instructions.

Some general notes:

- If you are rebuilding from source and wish to configure `GDB` to use the new Tk-based GUI, you must use the option `--enable-gdbtk` on the command line to `configure`.

Please report any problems you have with building under `progressive-96q1`.

Need to upgrade XLC to rebuild on AIX 3.2X

There is a reported problem in rebuilding the Developer's Kit using `IBM` native tools. (This problem does not crop up if you use `GCC` to rebuild the tools.)

On the `RS/6000`, `XLC` version 1.3.0.0 miscompiles 'jump.c'. `XLC` version 1.3.0.1 or later fixes this problem. You can obtain `XLC` version 1.3.0.2 by requesting `PTF 421749` from `IBM`.

This is not relevant for `AIX 4.1X`, where the compiler is newer.

See section "Rebuilding From Source" in *Rebuilding From Source*.

Use '--with-gnu-as' when configuring MIPS

If you rebuild the entire Developer's Kit from source, the top-level configuration files handle the following configuration detail for you automatically.

But if you rebuild the compiler *alone* for a `MIPS` target, we highly recommend that you specify `--with-gnu-as` on the command line for `configure`. This avoids an incompatibility between the `GNU` assembler and the `MIPS` assembler. The `MIPS` assembler does not support debugging directives, and `GCC` uses a special program, `mips-tfile`, to generate them. `GNU AS` parses the debugging directives directly, and does not require `mips-tfile`.

You should also specify `--with-stabs` on the command line to `configure`. This provides better debugging symbols, in particular for `C++`.

If you plan to use `GNU LD`, be sure to specify `--with-gnu-ld` when you rebuild on any platform for which the linker is available.

Rebuilding the tools under Solaris 2

If you wish to rebuild the tools from source on your `SPARC` system running Solaris 2, you can use either the original Solaris 2 native-development binaries from the Cygnus Support Developer's Kit or the unbundled compiler sold separately by Sun.

Beware! You might notice that there is a program called `'/usr/ucb/cc'` and be tempted to use it. Don't. This program is incompatible with the *real* compiler, which is in `'/opt/SUNWspro/bin/cc'`.

Multiple object code formats

As in previous releases, you can reconfigure the Developer's Kit tools to support more than one object format. See section "Rebuilding From Source" in *Rebuilding From Source*.

To add support for more object file formats (besides the format appropriate for the configured target), list the additional targets as arguments to the configure option `'--enable-targets'`, separated by commas. For example:

```
./configure --enable-targets=m68k-coff,i386-elf,decstation
```

To find out what targets are available, look in the file `'bfd/config.bfd'` in the source distribution.

To configure the Developer's Kit tools to support all available object formats, use `'--enable-targets=all'` rather than listing individual targets.

Problems fixed in this release

Here is a list of the problems we have fixed since the last Progressive release. We hope that you find it useful. (You can contact us at +1 415 903 1401 to inquire about the status of any problems.)

This information, as well as a list of all problems that have been reported to us that are still outstanding, is available in ASCII form from Cygnus Support. Contact us at support@cygnus.com or at the phone number above and ask for the ASCII list of known and/or fixed bugs for 96q1.

The following summaries of fixed bugs in the `progressive-96q1` release are organized by the reporting category—that is, by the software component, such as `gdb` or `g++`.

Each bug summary begins with the Cygnus Support Problem Report number. We consider a problem report *closed* only when the customer who reported the bug agrees the problem is solved.

Binary Utilities

- 6758** nm options incompatible with `lorder`
- 6795** `stbxxx` directory left behind when stripping a library

Problems rebuilding from source

- 6394** build questions

GNU C++ Compiler (G++)

- 2775** calling C++ functions from C
- 3097** static member initialization bug
- 3099** base class constructor order is incorrect
- 3617** Comparison of sign and unsigned produces incorrect result
- 3687** want error to read “component ‘processes’ is not a method and no operator()(...) defined on TYPE”
- 3778** `purify 2.1.0` with `gcc` (`cygnus-2.4.5-930716/progressive-930929`)
- 3872** `g++` can’t figure out how to use an user-defined operator `CAST` to cast to a descendant “`enum_domain.cc:93: cannot convert to a pointer type`”

- 4933** Can find the iostream.h file
- 5073** possible to inherit from class with private constructor
- 5087** virtual function before virtual destructor causes internal error
- 5098** Object size grows 15% with 94Q1 g++
- 5103** g++ doesn't notice that some pure virtuals has been redefined
- 5104** g++ complains about copy-constructor not returning a const value
- 5287** template problem
- 5370** delete dumps core with -fhuge-objects
- 5374** g++ gives erroneous error message with pointer to member and inheritance
- 5729** Passing overloaded static members as arguments to constructor does not work.
- 5730** Address of private static member function cannot be passed to constructor of friend class.
- 5732** Inline-function used before declared inline
- 5733** Error reported when protected member of virtual base is called.
- 5735** Illegal pointer assignment accepted by g++
- 5748** Default arguments must be specified in one place.
- 6189** virtual inheritance gives error for protected base member use (even if public).
- 6267** internal compiler error during g++ compilation
- 6556** Task-library
- 6825** empty component decl causes parse error
- 6826** g++ instantiates unnecessary templates
- 7099** g++ confuses object declaration for function prototype

GNU Assembler (AS)

- 2691** question about gas & Motorola syntax
- 2757** m68k-coff-as
- 2871** gas doesn't gracefully handle undefined referances
- 2975** gas produces the wrong offset
- 2997** one-pass warnings
- 3283** bfd app err
- 3483** the gas does not recognize membar instruction
- 3485** Gas core dumps while assembling trap_data_table.s
- 3597** assembler cannot create 64bit constants as described in SPARC V9
- 3604** gas dumps core for undefined value
- 3626** a branch to an external label+offset does not work
- 4182** program as got fatal signal 11
- 4280** one and retry instructions have reserved bits set
- 6152** new release has problem assembling
- 6509** Signal 11 in mips-idt-ecoff gcc
- 6750** .globl overrides .weak
- 6891** gas produces a bogus undefined extern ref
- 7015** filename length set to a maximum of 14 in m68k-coff-gdb
- 7148** unaligned access as a result of object file produced by gas

GNU C Compiler (GCC)

- 2812** sun3 gcc received was for a.out not COFF
- 6250** bug in gcc
- 6516** Bad code generated for ?: when compiled with -O2
- 6667** need unaligned pointers for specific variables
- 6739** Problem in gcc preprocessing
- 7112** compiler : bad variable initialization
- 7156** P3 tool-chain does not define __mips_single_float.

GNU Debugger (GDB)

- 2912** (xgdb) Where can I get a X front end for gdb?
- 4247** having trouble with gdb serial connection to target
- 4253** how to get source line #s with disassemble command in gdb
- 6471** target udi command under gdb

Miscellaneous GNU bug issues (gnats)

- 3454** Submitter ID should default per-site, not global default
- 3489** Makefile install rules/manual install instructions are wrong
- 4380** Error with reusing configuration with multiline ORGANIZATION
- 4490** *send-pr* buffer should behave more like *mail* buffer
- 5842** Wanted: Timely reminders also in feedback state

Problems installing the CDK

- 6608** Annoying warnings for stdio-functions
- 7111** release 95q2 : files missed.

GNU Linker (LD)

- 6269** ld dumps core with -lresolv
- 6459** ld loses relocation information when linking a large program
- 6594** ld reports: invalid number '-dy'
- 6871** GNU ld doesn't recognize a symbol in libresolv.a(gethostnamadr.o)
- 6889** ld gets SIGSEGV
- 6939** wilcards make od things happen
- 6959** error message "No symbols" could be more helpful
- 7145** size of debug info question
- 7164** ld crashes when etext is defined in shared object

Cygnus C Support Library (libc)

- 810** enhancements to genclass
- 7151** compile time warnings with -Wall when including Integer.h

Both Cygnus C Support Libraries (libc and libm)

- 6183** Missing __sread,swrite,sseek,sclose routines from fopen
- 6225** ../vfprintf.c:192: 'CVT_BUF_SIZE' undeclared
- 7071** PRO's tools bugs for Winbond's W89K PA-RISC Controller

Cygnus Support make

- 3517** make of progressive 930331 fails
- 6772** GNUmake debug mode turns on sometimes without specifying -d

Cygnus Support Problem Report Management System (PRMS)

- 3969** Detection of bad options in query-pr should be improved

Cygnus Support Problem Report Submission (send-pr)

- 145** change-requests for 'send-pr' (pilot-version)
- 2977** Release info gets erased
- 6607** send-pr reports error when .forward to program

Appendix A Specifying Names for Hosts and Targets

Your tape is labeled to indicate the host (and target, if applicable) for which the binaries in the distribution are configured. The specifications used for hosts and targets in the `configure` script are based on a three-part naming scheme, though the scheme is slightly different between hosts and targets.

Host names

The full naming scheme for hosts encodes three pieces of information in the following pattern:

architecture-vendor-os

For example, the full name for a Sun SPARCstation running SunOS 4.1.4 is

`sparc-sun-sunos4.1.4`

Warning: `configure` can represent a very large number of combinations of architecture, vendor, and operating system. There is by no means support for all possible combinations!

The following combinations refer to hosts supported by Cygnus Support. The OS versions shown are the versions under which this release was built. If you have any questions regarding compatibility, please feel free to contact Cygnus Support.

<i>canonical name</i>	<i>platform</i>
<code>sparc-sun-solaris2</code>	SPARCstation, Solaris 2.4
<code>sparc-sun-sunos4.1.4</code>	SPARCstation, SunOS 4.1.4
<code>mips-dec-ultrix</code>	DECstation, Ultrix 4.4
<code>rs6000-ibm-aix3.2.5</code>	IBM RS/6000, AIX 3.2.5
<code>mips-sgi-irix4</code>	SGI Iris, Irix 4.0.5H
<code>mips-sgi-irix5</code>	SGI Iris, Irix 5.3
<code>m68k-hp-hpux9</code>	HP 9000/300, HPUX B.09.00
<code>hppa1.1-hp-hpux9</code>	HP 9000/700, HPUX A.09.05
<code>hppa1.1-hp-hpux10</code>	HP 9000/700, HPUX B.10.01
<code>alpha-dec-osf2.0</code>	DEC Alpha, OSF/1 v2.0
<code>powerpc-ibm-aix4.1</code>	IBM PowerPC, AIX 4.1.3

`config.guess`

`config.guess` is a shell script which attempts to deduce the host type from which it is called, using system commands like `uname` if they are available. `config.guess` is remarkably adept at deciphering the proper configuration for your host; if you are building a tree to run on

the same host on which you're building it, we recommend *not* specifying the *hosttype* argument.

`config.guess` is called by `configure`; you need never run it by hand, unless you're curious about the output.

Appendix B Graphical User Interface for GDB

GDB in this release is configured to use the new Graphical User Interface. This interface is currently in a beta stage of development.

Under Unix, the GUI is called simply with `gdb`. The GUI is based on Tcl and the Tk windowing toolkit; a different system was used to develop the GUI under Windows (available soon), but the underlying structure is the same.

When running as a Unix program and using the X11-based interface, you must of course be using an X server and/or workstation, and your `DISPLAY` environment variable must be set correctly. If either of these is not true, then GDB still starts up, using the traditional command interface.

The exact layout and appearance of the windows depends upon the host system type. General behavior and layout is consistent across all platforms; omissions or restrictions on particular platforms, if not documented as unavoidable, should be considered bugs and reported.

All GDB windows have a common structure. Each window has an associated menu bar, which may be at the top of the window or perhaps elsewhere. Some of the menus and menu items in the menu bar are common to all GDB windows, while others are specific to particular types of windows. Below the menu bar is the working data area of the window. If the data is too large to display all at once, the data area shows scroll bars on its right and bottom sides. Below the data area are two optional features: a status/data line, and a button box.

For details on using GDB, see section “GDB Commands” in *Debugging With GDB*.

Getting Started

To launch the GDB GUI, simply type `gdb` on the command line. If you are in a windowing environment and your `DISPLAY` variable is set, the Command Window and Source Window appear. To suppress the GUI, use the `-nw` (*non-windowing*) option on the command line.

If you move the binaries to somewhere other than their original location, you will need to set the following three environment variables:

```
TCL_LIBRARY
TK_LIBRARY
GDBTK_FILENAME
```

`TCL_LIBRARY` is the location of tcl code that tcl expects to be able to find, while `TK_LIBRARY` is the same for tk.

`GDBTK_FILENAME` is the pathname to the tcl code that actually defines the GDB graphical interface.

Menus

File Menu

The standard file menu provides operations that affect the overall state of GDB, mainly file operations, but other things as well.

File... Lets you set the combined executable and symbol file that GDB will use. (Like the gdb command 'file'.)

Target... Brings up a dialog that you can use to connect GDB to a target program. The dialog is described in more depth later. (Like the gdb command 'target'.)

Edit... Starts up an editor to modify the source file being displayed.

Exec File... Lets you set the executable file that GDB will use. (Like the gdb command 'exec-file'.)

Symbol File... Lets you set the symbol file that GDB will use. (Like the gdb command 'symbol-file'.)

Add Symbol File... Lets you add additional symbol files. (Like the gdb command 'add-symbol-file'.)

Core File... Lets you set the core file that GDB will use. (Like the gdb command 'core-file'.)

Shared Libraries... (Like the gdb command 'sharedlibrary'.)

Quit Quits GDB. (Like the gdb command 'quit'.)

Options Menu

The Options Menu is different for each window, showing the viewing options available within that window. In the Source Window, for example, one of the options is whether to display line numbers.

Window Menu

The Window Menu allows access to all the windows available in GDB. The first part of the menu lists all of the predefined individual windows. If the window exists already, its item will be marked as such; selecting the item will cause the window to be put in front if it is obscured. If the window does not exist, then it will be created.

The second part of the menu lists additional windows that you may have created, such as source windows or variable displays.

Selections on this menu include:

- Command
- Source
- Assembly
- Registers
- Variables
- Files
- *any other windows you have open*

Help Menu

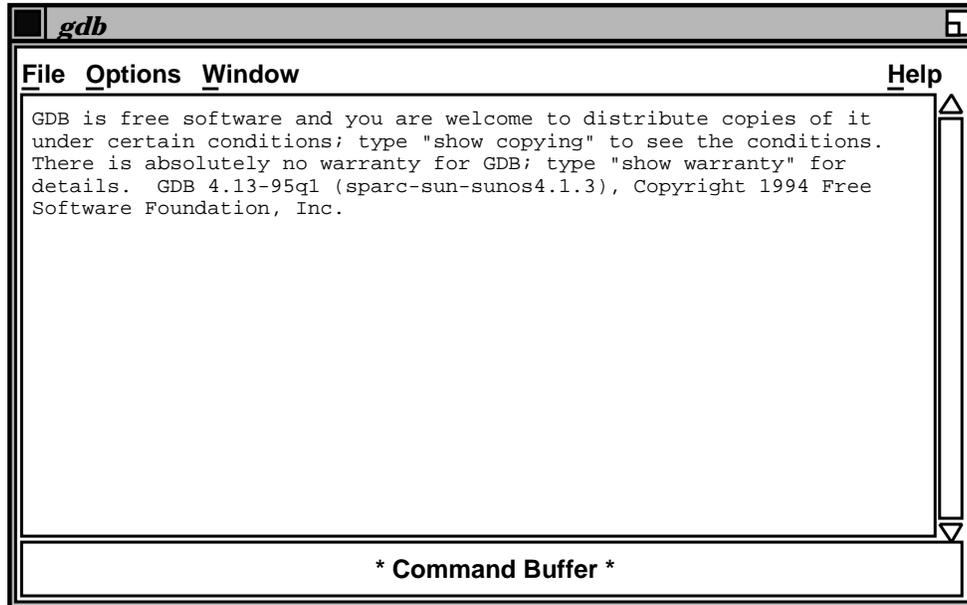
The Help Menu includes access to GDB's online help.

Windows

Command Window

The command window provides access to the standard GDB command interpreter. In nearly all cases, commands typed into this window behave exactly as for a non-windowing GDB.

Note that not all changes to GDB are reflected in this window. For instance, if you type a 'step' command, then click on the 'step' menu item in the source window, then type another 'step' command, the command buffer shows only two steps, although you have actually done three. GDB places an ellipsis (...) in the command buffer when operations in other windows are done, as a reminder that the command buffer is incomplete. The command window has no status line or button box.



(This example screen is the initial Command Window from the Unix version of the GDB GUI.)

Files Window

The Files Window lists all of the files that were used to build the executable.

Clicking in the bar in the left margin expands/contracts the display of included files and symbols defined by the file.

Source Window

A source window displays a single file of source code.

The left margin includes an indicator for the current PC, breakpoints and potential breakpoints, and (optionally) line numbers.

Rebuilding the GDB GUI

If you are rebuilding the Cygnus Developer's Kit (or GDB by itself), use the '--enable-gdbtk' option to configure GDB to use the new GUI.

For more information on rebuilding from source, see the manual *Rebuilding From Source*. Also, be sure to check "Problems rebuilding from source", page 13, for some known problems in rebuilding this release.