

The UDUNITS-2 Package

Steven R. Emmerson

Copyright 2014 University Corporation for Atmospheric Research and contributors. All rights reserved.

This software was developed by the Unidata Program Center of the University Corporation for Atmospheric Research (UCAR) <<http://www.unidata.ucar.edu>>.

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 names of the development group, the copyright holders, nor the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission. 4) This license shall terminate automatically and you may no longer exercise any of the rights granted to you by this license as of the date you commence an action, including a cross-claim or counterclaim, against the copyright holders or any contributor alleging that this software infringes a patent. This termination provision shall not apply for an action alleging patent infringement by combinations of this software with other software or hardware.

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 CONTRIBUTORS 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 WITH THE SOFTWARE.

Table of Contents

1	Introduction	1
2	How This Package Differs from the Original UDUNITS Package	2
3	Installing this Package	3
3.1	Prerequisites for This Package.....	3
3.2	Obtaining this Package.....	3
3.3	Unix Installation Instructions	3
3.3.1	Autoconf-based Installation.....	3
3.3.2	CMake-based Installation.....	4
3.4	Windows Installation Instructions	4
4	Unit Library	5
5	Unit Utility	6
6	The Units Database	7
7	Support for this Package	8
	Index	9

1 Introduction

The UDUNITS-2 package provides support for units of physical quantities. Its three main components are: 1) [Section “a C library” in `udunits2lib`](#) for units of physical quantities; 2) [Section “a utility” in `udunits2prog`](#); for obtaining the definition of a unit and for converting numeric values between compatible units; and 3) an [Chapter 6 \[extensive database of units\]](#), [page 7](#).

2 How This Package Differs from the Original UDUNITS Package

The UDUNIT-2 package differs from the UDUNITS-1 package in the following ways:

- **Support for non-ASCII characters:** The UDUNITS-1 package only supports the ASCII character set. The UDUNITS-2 package supports the following character sets: ASCII, ISO 8859-1 (Latin-1), and the UTF-8 encoding of ISO 10646 (Unicode). This means that unit string specifications like "F" are now supported (your viewer must support UTF-8 to display this string correctly).
- **Support for logarithmic units:** The unit string specification "0.1 lg(re 1 mW)" specifies a deciBel unit with a one milliwatt reference level. Such units are fully integrated into the package and all meaningful operations and conversions are supported.
- **Persistent value converters:** It is now possible to obtain a *converter* data-object, which can be used to convert numeric values in one unit to numeric values in another, compatible unit. The values can be `float`, `double`, or one-dimensional arrays of `floats` or `doubles`.
- **Improved API:** Due to the above changes, it was not possible to keep the application programming interface (API) of the UDUNITS-1 package. The new interface, however, is easily understood and easy to use. To support backward compatibility, the package does contain a version 1 C API, which uses the version 2 library.
- **XML unit database:** The unit database is encoded using human-readable XML rather than a custom format. The XML parser included in the package supports an *<import>* element to allow easy and convenient customization.
- **No Fortran or Perl API:** Interfaces for these languages have not yet been created. Contact support-udunits@unidata.ucar.edu if you are interested in helping to create these interfaces.

One thing that has not changed is that almost all unit string specifications understood by the UDUNITS-1 package are also understood by the UDUNITS-2 package. One exception is the symbol `g`, which in version 1 of the package was associated with standard free fall (a unit of acceleration) but which is associated with the unit gram in version 2 of the package.

3 Installing this Package

3.1 Prerequisites for This Package

The Expat library and header-files must be installed. Look for the library "libexpat" and the header-file "expat.h".

If this package is to be tested (which is optional), then the CUnit library and header-files must be installed. Look for the library "libcunit" and the header-files "CUnit/CUnit.h" and "CUnit/Basic.h".

If this package is to be built on a Windows system, then the CMake and MinGW packages must be installed. Look for the utility "cmake" and the directory/folder "MinGW".

3.2 Obtaining this Package

Get the source for this package from its download-page at <ftp://ftp.unidata.ucar.edu/pub/udunits/> and unpack it in an appropriate place. For example,

```
su -
cd /usr/local/src
wget -O - ftp://ftp.unidata.ucar.edu/pub/udunits/udunits-2.2.28.tar.gz |
pax -rz
```

3.3 Unix Installation Instructions

3.3.1 Autoconf-based Installation

Short instructions:

```
./configure [--prefix=root] [--disable-shared] [CC=path]
make
make check           # optional; requires CUNIT installation
make install         # also installs INFO documentation
make install-html install-pdf # optional
make clean
```

By default, the installation root is /usr/local.

Long instructions:

1. Go to the top-level source-directory of this package, e.g.,


```
cd udunits-2.2.28
```
2. If necessary, clean-up from a previous installation attempt by making the `distclean` target using the `make` utility from step 2:


```
make distclean
```
3. Execute the `configure` script. Specify the installation prefix, the compiler from the previous step, and any required option. For example, if you are on an HP-UX system and want to use the `/bin/c89` compiler, create a static library, and install under `/opt`, then the following command is appropriate:

```
./configure --prefix=/opt --disable-shared CC=/bin/c89
```

The option `--disable-shared` causes the build process to create a static library only: a sharable library is not created. This option is necessary if the `libtool` utility that's included in this package can't build a sharable library using the given compiler.

The default installation prefix is `/usr/local` and the default compiler is `gcc`.

4. Build this package by making the default target using the `make` utility from step 2:

```
make
```

5. If you wish to verify that this package works correctly, then make the `check` target using the `make` utility from step 2:

```
make check
```

This step is only effective if the `configure` script found an installed CUNIT unit-testing package. If that package wasn't found, then the above command will quickly exit without testing this package.

6. Install the [Section "library" in `udunits2lib`](#), [Section "utility" in `udunits2prog`](#), header-files, units-database, and INFO documentation files by making the `install` target using the `make` utility from step 2:

```
make install
```

7. If desired, install the HTML and PDF documentation files by making one or more of the following targets using the `make` utility from step 2:

```
make install-html install-pdf
```

8. (Optional) Clean up by making the `clean` target using the `make` utility from step 2:

```
make clean
```

3.3.2 CMake-based Installation

```
mkdir build
cd build
cmake ..
make all [test] install [install_test]
```

where the arguments in square brackets are optional.

3.4 Windows Installation Instructions

Currently, only a 32-bit [MinGW](#)-based installation is supported:

```
cmake -G "MinGW Makefiles"
cmake --build . [-DCMAKE_INSTALL_PREFIX=dir] -- all [test] install [install_test]
```

where arguments in square brackets are optional.

The default installation-prefix is `"C:\Program Files (x86)\udunits-2.minor.bug"`, where `minor` and `bug` are the minor version and bug-fix level, respectively.

4 Unit Library

See [Section “UDUNITS Library”](#) in `udunits2lib`, for details on this package’s unit library.

5 Unit Utility

See [Section “UDUNITS Utility”](#) in `udunits2prog`, for details on this package’s unit utility.

6 The Units Database

The database for the UDUNITS-2 package comprises one XML file containing unit prefixes and four XML files containing unit definitions:

- SI unit prefixes
- SI base units
- SI derived units
- Units accepted for use with the SI
- Non-SI units

7 Support for this Package

The home-page for this package can be found at <http://www.unidata.ucar.edu/software/udunits/>.

Bug reports should be sent to support-udunits@unidata.ucar.edu.

Index

D

database..... 7

I

installation..... 3

Introduction..... 1

L

Library, unit..... 5

O

obtaining this package..... 3

P

package, support..... 8

prerequisites for this package..... 3

S

support..... 8

U

Utility, unit..... 6