NAME
      build-all — build software distributions on remote systems in parallel

SYNOPSIS
      build-all [−−**?**] [−−**all** *target(s)*] [−−**check** *target(s)*] [−−**configure** *options*] [−−**help**]
               [−−**logdirectory** *dir*] [−−**on** *[user@]host[:dir][,envfile]* . . .] [−−**source** *dir(s)*]
               [−−**userhosts** *file(s)*] [−−**version**] package(s)

OPTIONS
      build-all options can be prefixed with either one or two hyphens, and can be abbreviated to any unique pre-
      fix. Thus, −**v**, −**ver**, and −−**version** are equivalent.

      To avoid confusion with options, if a filename begins with a hyphen, it must be disguised by a leading abso-
      lute or relative directory path, e.g., */tmp/-foo* or *./-foo*.

      −−**all** *target(s)*   Specify the default **make**(1) targets to build the package [default: empty].

      −−**?**   Same as −−**help**.

      −−**check** *target(s)*   Specify the default **make**(1) targets to test the package [default: *check*].

      −−**configure** *options*   Specify any options needed by the **configure**(1) script.

      −−**help**   Display a brief help message on *stdout*, giving a usage description, and then termi-
                                    nate immediately with a success return code.

      −−**logdirectory** *dir*   Specify an alternate log file directory. By default, logs are created in a subdirectory
                                    named *logs/package-x.y.z* for package *package-x.y.z* in the directory in which the
                                    package archive is found. If the directory cannot be created or written to, a default
                                    log directory is chosen and reported.

      −−**on** *[user@]host[:dir][,envfile]* . . .
                                    Specify an alternate list of hosts on which to build the packages. The list must be
                                    quoted if it contains spaces or other shell metacharacters. This option suppresses
                                    reading the host list from the *$HOME/.build/userhosts* initialization file, allowing
                                    builds to be restricted to a particular set of hosts.

                                    In each host specification, the username may be omitted if it is the same as on the
                                    current host. The hostname is mandatory. The directory in which packages are
                                    unbundled and built is given by *dir*: a default directory is chosen and reported if that
                                    value is omitted. The optional environment file, *envfile*, on the initiating host (*not*
                                    the remote host), provides settings of environment variables for **configure**(1), in the
                                    form of *key=value* assignments.

                                    This option may be given any number of times.

      −−**source** *dir(s)*   Specify an additional list of directories to search for source archives. The list must
                                    be quoted if it contains spaces or other shell metacharacters. That list will be
                                    searched before the directories in the *$HOME/.build/directories* list, or the built-in
                                    list.

                                    This option may be given any number of times.

      −−**userhosts** *file(s)*   Specify an alternate list of userhosts files to replace the default initialization file,
                                    $HOME/.build/userhosts. The list must be quoted if it contains spaces or other shell
                                    metacharacters.

                                    If a specified userhosts file is not readable as given, it is looked for in
                                    $HOME/.build,

                                    This option may be given any number of times.

                                    Settings from the −−**on** option override this option, so the two should be considered
                                    mutually exclusive.

This option makes it convenient to create sets of build hosts grouped by compiler, platform, or package, to accommodate packages that are known to build only in certain limited environments.

−−**version**   Display the program version number and release date on *stdout*, and then terminate immediately with a success return code.

DESCRIPTION

build-all provides for automated software builds on one or more hosts, in parallel, with build logs collected in package-specific directories on the initiating host, using the secure shell to communicate with the remote hosts.

The log directory is always reported on *stdout*, and in an email message.

Log files are named with the package name, version, build host, and an ISO 8601:2000 odometer-style time stamp, like this: *package-x.y.z.hostname.yyyy.mm.dd.hh.mm.ss.log*.

Packages should conform to GNU conventions for software packaging: an archive file *package-x.y.z.tar.gz* unbundles into a directory named *package-x.y.z* in which there is an optional **configure**(1) script, and a *Makefile*. Running **make**(1) in that directory should build the package, and **make check** (overridable with the −−**check** *target(s)* option) should run validation tests.

build-all recognizes archives in these formats: *.jar*, *.tar*, *.tar.bz2*, *.tar.gz*, *.tar.Z*, and *.zip*. It is the user's responsibility to ensure with suitable shell startup files that on each remote host, GNU **tar**(1) and **unzip**(1) are found first in the search path, since they are needed to unbundle packages.

Installation after builds is *not* automated; it is imperative to examine the build logs before deciding whether it is safe to issue **make install** in a remote build directory.

FILES

In this section, *$REMOTEHOME* refers to the home directory on a remote build host.

Comments are supported in all of these files: they run from sharp (#) to end of line.

$REMOTEHOME/.build/begin

Commands to be executed in the context of the login shell on the remote host before the build begins. This file can be used to augment the search path, *PATH*, for shells like **ksh**(1) and **sh**(1) that do not provide for startup files for other than login sessions. Any output produced will be recorded near the start of the build log file.

$HOME/.build/directories   The list of local directories to search for package archive files. If that file cannot be found, a default list of source directories is provided.

$REMOTEHOME/.build/end   Commands to be executed in the context of the login shell on the remote host after the build ends. Any output produced will be recorded near the end of the build log file.

$HOME/.build/userhosts   The default list of build hosts, with entries like this:

            jones@example.com:/local/build,$HOME/.build/c99

The username, build directory, and environment file are optional. Except for the remote build directory, all files listed are on the initiating host.

envfile   Environment files contain settings of environment variables for **configure**(1) on the remote systems. Filenames are arbitrary: they are set in entries in the *$HOME/.build/userhosts* file or similar files specified by the −−**userhosts** option, or else in command-line −−**on** options. For example, an environment file with

            CC=c99
            CFLAGS="-O3 -D_POSIX_SOURCE"
            CXX=CC

CXXFLAGS="-O3 -D_POSIX_SOURCE"

specifies the C and C++ compilers, and compiler optimization flags, to be used for the build.

EXAMPLES

Build two packages everywhere:

```
build-all coreutils-5.0.0 gawk-3.1.2
```

Build package on a specific host:

```
build-all --on loaner.example.com gnupg-1.2.1
```

Build package from nonstandard location:

```
build-all --source $HOME/work butter-0.3.7
```

Build package in POSIX compilation environment:

```
build-all --userhosts posixhosts numlib-3.2.1
```

SEE ALSO

autoconf(1), **autoheader**(1), **automake**(1), **configure**(1), **jar**(1), **ksh**(1), **make**(1), **scp**(1), **sh**(1), **ssh**(1), **tar**(1), **unzip**(1).