

```
##### <schweikh@FreeBSD.org>
#####: 43126
##### © 2002,2003,2004,2008 #####
```

```
#####,#####,##### #####,##### ## ##### ## #—
##### ##### ## ##### ## ##### #####
##### ## ## ##### ##### ##/# ##### #####.
```

2013#11#07 ## #####.

```
#####  
#####.  
#####  
#####  
#####  
#####  
#####  
#####  
#####  
#####  
#####  
#####
```

1. #####	2
2. ### ##### # (###) #### #####?	2
3. #####	3
4. ##### #: #####	4
5. ##### #: #####	10
6. #####	13
7. #####	13
8. ### #####	14





##### #: #####

- [illegible]

4. ##### ###: ##### #####

#####  
#####  
#####  
#####  
#####  
#####.

```
#### ##### ##### ##### ## stage_1.sh #### ##### ##### ##### #####,
####
```

```
# ./stage_1.sh default
```

```
##### stage_1.conf.default ##### stage_1.log.default.
```

```
##### stage_1.conf.default. #####  
#####  
#####  
#####  
#####  
#####, create_file_systems, create_etc_fstab, copy_files ## all_remaining_customization  
(#####:  
stage_1.sh).
```

### ##### ## ##### ###:

- #####.

[illegible]

- ##### #####.

#####  
#####  
#####  
#####

#####

#####, ### ### #####  
## ### ### #####, ##### SIGSYS,  
Bad system call, #####  
#####/#####5.8.

##### stage\_1.sh #####  
##### make installworld installkernel, #####:

- #####
- ##### make buildworld
- ##### make buildkernel KERNCONF=whatever

#### stage\_1.sh ####  
#####  
/usr/src, mergemaster ####  
#### running ##### (####  
#### -STABLE #### -CURRENT. ####  
##### mergemaster #####  
##### /usr/src ####

### stage\_1.sh #####  
#####  
### set -e, #####  
##### stage\_1.conf.default #####

## stage\_1.sh ## mergemaster. #####  
#####

\*\*\* Comparison complete  
\*\*\* Savingmtree database for future upgrades

Do you wish to delete what is left of -/var/tmp/temproot.stage1? [no] **no**

##### no ##  
# ## /var/tmp/temproot.stage1  
#####

##### mergemaster ##### login.conf  
#####

\*\*\* You chose the automatic install option for files that did not  
exist on your system. The following were installed for you:  
-/newroot/etc/defaults/rc.conf  
~...  
-/newroot/COPYRIGHT

\*\*\* You installed a new aliases file into -/newroot/etc/mail, but

```
the newaliases command is limited to the directories configured
in sendmail.cf. Make sure to create your aliases database by
hand when your sendmail configuration is done.
```

```
*** You installed a login.conf file, so make sure that you run
-/usr/bin/cap_mkdb -/newroot/etc/login.conf
to rebuild your login.conf database
```

```
Would you like to run it now? y or n [n]
```

```
# This file: stage_1.conf.default, sourced by stage_1.sh.
#
# $Id: stage_1.conf.default,v 1.5 2011-05-14 20:44:31 hrs Exp $
# $FreeBSD: head/en_US.ISO8859-1/articles/fbsd-from-scratch/stage_1.conf.default 38826 -
2012-05-17 19:12:14Z hrs $

# Root mount point where you create the new system. Because it is only
# used as a mount point, no space will be used on that file system as all
# files are of course written to the mounted file system(s).
DESTDIR="/newroot"

# Where your src tree is.
SRC="/usr/src"

# Where your obj is.
MAKEOBJDIRPREFIX="/usr/obj"

# Your kernel config name as from make buildkernel KERNCONF=...
KERNCONF="HAL9000"

# Your target architecture as used for make buildworld TARGET=...
# If you did not specify a TARGET when building world, it defaulted
# to the build architecture (run -"uname --m" to find out if you are unsure).
TARGET="i386" # amd64 arm i386 ia64 mips pc98 powerpc sparc64

# Available time zones are those under -/usr/share/zoneinfo.
TIMEZONE="Europe/Berlin"

#
# The create_file_systems function must create the mountpoints under
# DESTDIR, create the file systems, and then mount them under DESTDIR.
#
create_file_systems () {
    # The new root file system. Mandatory.
    # Change DEVICE names.
    DEVICE="/dev/daXYZsl1a
mkdir --m 755 -p ${DESTDIR}
```

#####

```
chown root:wheel ${DESTDIR}
newfs -U ${DEVICE}
mount --o noatime ${DEVICE} ${DESTDIR}

# Additional file systems and initial mount points. Optional.
DEVICE=/dev/daXYZs1e
mkdir --m 755 --p ${DESTDIR}/var
chown root:wheel ${DESTDIR}/var
newfs -U ${DEVICE}
mount --o noatime ${DEVICE} ${DESTDIR}/var

DEVICE=/dev/daXYZs1e
mkdir --m 755 --p ${DESTDIR}/usr
chown root:wheel ${DESTDIR}/usr
newfs -U ${DEVICE}
mount --o noatime ${DEVICE} ${DESTDIR}/usr
}

#
# The create_etc_fstab function must create an fstab matching the
# file systems created in create_file_systems.
#
create_etc_fstab () {
    cat <<EOF >${DESTDIR}/etc/fstab
# Device      Mountpoint    FStype  Options          Dump Pass#
/dev/da0s1b    none          swap    sw               0 0
/dev/da1s1b    none          swap    sw               0 0
/dev/da2s2b    none          swap    sw               0 0
/dev/da3s2b    none          swap    sw               0 0
/dev/da0s1a    -/            ufs     rw,noatime       1 1
/dev/da0s1e    -/var         ufs     rw,noatime       1 1
/dev/da2s1e    -/usr         ufs     rw,noatime       1 1
/dev/vinum/Share -/share       ufs     rw,noatime       0 2
/dev/vinum/home -/home        ufs     rw,noatime       0 2
/dev/vinum/ncvs -/home/ncvs   ufs     rw,noatime       0 2
/dev/vinum/ports -/usr/ports   ufs     rw,noatime       0 2
/dev/ad1s1a    -/flash       ufs     rw,noatime       0 0
/dev/ad0s1     -/2k          ntfs    ro,noauto        0 0
/dev/ad0s6     -/linux       ext2fs  ro,noauto        0 0
#
/dev/cd0       -/cdrom       cd9660  ro,noauto        0 0
/dev/cd1       -/dvd         cd9660  ro,noauto        0 0
proc          -/proc        procfs  rw               0 0
linproc       -/compat/linux/proc linprocfs rw               0 0
EOF
    chmod 644 ${DESTDIR}/etc/fstab
    chown root:wheel ${DESTDIR}/etc/fstab
}

#
# The copy_files function is used to copy files before mergemaster is run.
#
copy_files () {
```

##### #: #####

```
# Add or remove from this list at your discretion. Mostly mandatory.
for f in \
  ~/.profile \
  /etc/devd.conf \
  /etc/devd.rules \
  /etc/exports \
  /etc/group \
  /etc/hosts \
  /etc/inetd.conf \
  /etc/ipfw.conf \
  /etc/make.conf \
  /etc/master.passwd \
  /etc/nsswitch.conf \
  /etc/ntp.conf \
  /etc/printcap \
  /etc/profile \
  /etc/rc.conf \
  /etc/resolv.conf \
  /etc/src.conf \
  /etc/sysctl.conf \
  /etc/ttys \
  /etc/mail/aliases \
  /etc/mail/aliases.db \
  /etc/mail/hal9000.mc \
  /etc/mail/service.switch \
  /etc/ssh/*key* \
  /etc/ssh/*_config \
  /etc/X11/xorg.conf \
  /var/cron/tabs/* \
  /root/.profile \
  /boot/*.bmp \
  /boot/loader.conf \
  /boot/device.hints -; do
  cp --p ${f} ${DESTDIR}${f}
done
}

#
# Everything else you want to tune in the new system.
# NOTE: Do not install too many binaries here. With the old system running and
# the new binaries and headers installed you are likely to run into bootstrap
# problems. Ports should be compiled after you have booted in the new system.
#
all_remaining_customization () {
  # Without the compat symlink the linux_base files end up on the root fs:
  cd ${DESTDIR}
  mkdir --m 755 usr/compat; chown root:wheel usr/compat; ln --s usr/compat
  mkdir --m 755 usr/compat/linux;   chown root:wheel usr/compat/linux
  mkdir --m 555 usr/compat/linux/proc; chown root:wheel usr/compat/linux/proc
  mkdir --m 755 boot/grub;          chown root:wheel boot/grub
  mkdir --m 755 linux 2k;           chown root:wheel linux 2k
  mkdir --m 755 src;                chown root:wheel src
  mkdir --m 755 share;              chown root:wheel share
}
```



#####

```
mkdir -m 755 dvd cdrom flash;      chown root:wheel dvd cdrom flash
mkdir -m 755 home;                  chown root:wheel home
mkdir -m 755 usr/ports;              chown root:wheel usr/ports
```

```
# Create the ntp and slip log files.
touch ${DESTDIR}/var/log/ntp ${DESTDIR}/var/log/slip.log
```

```
# Make -usr/src point to the right directory. Optional.
# Note: some ports need part of the src tree, e.g. emulators/kqemu,
# sysutils/lsof, sysutils/fusefs, -...
cd ${DESTDIR}/usr
if test -"${SRC}" -!= -usr/src; then
    rmdir src; ln -s ${SRC}
fi
if test -"${MAKEOBJDIRPREFIX}" -!= -usr/obj; then
    rmdir obj; ln -s ${MAKEOBJDIRPREFIX}
fi
```

```
# My personal preference is to symlink tmp --> var/tmp. Optional.
cd ${DESTDIR}; rmdir tmp; ln -s var/tmp
```

```
# Make spooldirs for the printers in my -/etc/printcap.
cd ${DESTDIR}/var/spool/output/lpd; mkdir -p as od ev te lp da
touch ${DESTDIR}/var/log/lpd-errs
```

```
# If you do not have -/home on a shared partition, you may want to copy it:
# mkdir -p ${DESTDIR}/home
# cd -/home; tar cf -- -. -| (cd ${DESTDIR}/home; tar xpvf --)
}
```

```
# vim: tabstop=2:expandtab:shiftwidth=2:syntax=sh:
# EOF $RCSfile: stage_1.conf.default,v $
```

##### stage\_1.conf.default .

#####

- #####.
- #####.
- #####.
- #####, ##### /etc/ttys ### inetd.

#####  
## #####. ## #####  
#####  
#####  
#####  
#####.

5. ##### #: #####



```
##### (#####) #####  
#####, #####. ## ###, stage_2.sh  
##### pkg_add #####. # #####  
#####. #####  
#####
```

```
# ./stage 2.sh default
```

```
#####  
#####  
#####  
#####  
#####  
#####  
#####  
#####
```

```
www.mozilla make WITHOUT_MAILNEWS=yes WITHOUT_CHATZILLA=yes install
```

```
## ##### ## ## ##### ##### ##### ## ## ## ## ## ##### ##
##### make #####:
```

```
java jdk16      echo true > files/license.sh; make install BATCH=yes < -/dev/null
print acroread8 yes accept -| make install PAGER=ls
x11-fonts gnu-unifont make install && mkfontdir -/usr/local/lib/X11/fonts/local
news inn-stable CONFIGURE_ARGS="--enable-uucp-rnews --enable-setgid-inews" make install
```

```
#####  
#####.  
#####  
##### CONFIGURE_ARGS.  
##### Makefile  
#####.  
#####
```

```
news inn-stable make CONFIGURE_ARGS="--enable-uucp-rnews ---enable-setgid-inews" install
```

```
#####  
#####
```

#####

```
## ##### ## # # # # # # # # # # # # # # ,#.### ### ##  
## # # # # # # # # # # # # # # # # # # # # # # # # # # .# ####  
#,#### # # # # # (#)##### # # # # # # # # # # # # # # #  
### # # # # # # stage_2.sh##### # # # # # # # # # # # #,  
## #####.
```

```
##### ## stage_2.conf.default. ##### LOGDIR/category+port ## ##### ##  
## #####.
```

```
# vim: syntax=sh
# $Id: stage_2.conf.default,v 1.4 2008-12-03 21:59:51 schweikh Exp $
# $FreeBSD: head/en_US.ISO8859-1/articles/fbsd-from-scratch/stage_2.conf.default 38826 -
2012-05-17 19:12:14Z hrs $
ports-mgmt portaudit
devel ccache
shells zsh
devel gettext
archivers unzip
archivers zip
security sudo
x11 xorg
x11-servers xorg-server
x11-fonts xorg-fonts-100dpi
x11-fonts xorg-fonts-75dpi
x11-fonts xorg-fonts-miscbitmaps
x11-fonts xorg-fonts-truetype
x11-fonts xorg-fonts-type1
x11-fonts gnu-unifont make install && mkfontdir -/usr/local/lib/X11/fonts/local
x11-fonts urwfonts
x11-fonts webfonts
x11-toolkits open-motif
x11-wm ctwm
x11 wdm
security openssh-askpass
astro xplanet
astro xephem
editors vim
print ghostscript8
print psutils-a4
print a2ps-a4
print gv
print transfig
print teTeX
print cups-base
emulators linux_base-fc6
print acroread8 yes accept -| make install PAGER=ls
java jdk16 echo true > files/license.sh; make install BATCH=yes < -/dev/null
www apache22
www amaya
www firefox3
www checkbot
www p5-HTML-Parser
```

```
www validator
www mplayer-plugin
math p5-Math-Combinatorics
math p5-Bit-Vector
graphics evince
graphics xfig
graphics xv
graphics gphoto2
multimedia xawtv
lang expect
lang gawk
lang python
news tin
net freebsd-uucp
net cvsup-without-gui
net rsync
ftp wget
textproc ispell
german ispell-neu
german ispell-alt
textproc docproj
sysutils samefile
sysutils smartmontools
sysutils pstree
sysutils cdrtools
sysutils dvd+rw-tools
sysutils grub
sysutils lsof
devel subversion-freebsd
devel bcc
devel ddd
devel gindent
devel ctags
devel ElectricFence
devel strace
devel perltidy
mail procmail
mail metamail
mail mutt-devel
ports-mgmt portupgrade
news inn CONFIGURE_ARGS="--enable-uucp-rnews ---enable-setgid-inews" make BATCH=yes -
install < -/dev/null
misc figlet-fonts
security gpa
mail spamoracle
textproc rman
multimedia mplayer
multimedia mplayer-fonts
multimedia acidrip
multimedia ogle
multimedia ogle-gui
audio pacpl
audio p5-CDDDB_get
```

audio cowbell  
shells bash  
editors openoffice.org-3-RC  
java eclipse  
java netbeans

6. #####

```
#####  
#####  
##### stage_2.sh  
#####  
#####  
#####
```

```
# ##### Makefile #####
#####:
```

```
# make --f stage_3.mk target
```

```
## ##### stage_2.sh ##### ##### stage_3.mk ##### ##### ##### #####-
##, ##### ## ##### ## ## # ##### ##### ## ##### ## ##### ## ##
## #####.
```

7. #####

```
#####  
##### make BATCH=YES install.#####  
##### yes#####.#####  
#####  
#####  
##### (#.#. yes | make install.#####  
#####  
#####  
#####/#####8#####16.
```

```
##### ## ##### # ##### # ##### # #####.## ##### # # #  
##### # # # # ##### # ##### # # # ##### # # # # #. # # # # #  
# # # # # /etc/group, ## /etc/passwd ### # # # # #.### # # # # #  
# # # # #.##### # # # # # # # # # # # # # # # # # # # # # #  
##### # # # # #,### # # # # # # # # # #.## # # # # # # # # # #  
##### # # (# # #####)### # # # # # # # # # #mergemaster  
# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #  
##### # # # #.##### ,mergemaster # # # # # # # # # # # # # # #  
# # # # #.### # # # # # # # # # # # # # # # # # # # # # #
```

```
## ## ##### ## ## ## ## ## ##### ## ##### ## ## ##-  
## #####. ## ##### ## ## ## ## ## ## ## ## ## ## ##  
##### ## ## ## ## ## ## ## ## stage_3.mk ## ##### ## ## ##  
# ## ## ##, #. ## ##### httpd.conf ## ##### config_apache ####
```

```
## ### ##### # ### ##### ## ##### cp /usr/local/etc/apache2/
httpd.conf httpd.conf.
```

```
##### 7-CURRENT ## 7-CURRENT  
## 8-CURRENT ## 8-CURRENT, #.# ##### 8-CURRENT ####  
7-CURRENT ##### ## #### #.# ## ### ##### ## #####  
#####  
#####  
#####  
##### STABLE #####  
##### (##### # #### ##) 
```

#####

```
#!/bin/sh
#
# stage_1.sh -- FreeBSD From Scratch, Stage 1: System Installation.
#      Usage: -./stage_1.sh profile
#      will read profile
#      and write -./stage_1.log.profile
#
# Author:   Jens Schweikhardt
# $Id: stage_1.sh,v 1.7 2008-12-11 19:48:21 schweikh Exp $
# $FreeBSD: head/en_US.ISO8859-1/articles/fbsd-from-scratch/stage_1.sh 38826 2012-05-17 -
19:12:14Z hrs $
```

```
# Prerequisites:
#
# a) Successfully completed "-make buildworld" and "-make buildkernel"
# b) Unused partitions (at least one for the root fs, probably more for
#    the new -usr and -var, to your liking.)
# c) A customized profile file.
```

#####

```
echo -"usage: stage_1.sh profile" 1>&2
exit 1
fi

# ----- #
# Step 1: Create an empty directory tree below $DESTDIR.
# ----- #

step_one () {
    create_file_systems
    # Now create all the other directories. Mandatory.
    cd ${SRC}/etc; make distrib-dirs DESTDIR=${DESTDIR} TARGET=${TARGET}
}

# ----- #
# Step 2: Fill the empty -/etc directory tree and put a few files in -/.
# ----- #

step_two () {
    copy_files

    # Delete mergemaster's temproot, if any.
    TEMPROOT=/var/tmp/temproot.stage1
    if test -d ${TEMPROOT}; then
        chflags --R 0 ${TEMPROOT}
        rm -rf ${TEMPROOT}
    fi
    export MAKEDEVPATH="/bin:/sbin:/usr/bin"
    mergemaster -i --m ${SRC}/etc --t ${TEMPROOT} --D ${DESTDIR}
    cap_mkdb ${DESTDIR}/etc/login.conf
    pwd_mkdb -d ${DESTDIR}/etc --p ${DESTDIR}/etc/master.passwd

    # Mergemaster does not create empty files, e.g. in -/var/log. Do so now,
    # but do not clobber files that may have been copied with copy_files.
    cd ${TEMPROOT}
    find - . --type f -| sed -'s,^\./,,' -|
    while read f; do
        if test -r ${DESTDIR}/${f}; then
            echo -"${DESTDIR}/${f} already exists; not copied"
        else
            echo -"Creating empty ${DESTDIR}/${f}"
            cp --p ${f} ${DESTDIR}/${f}
        fi
    done
    chflags --R 0 ${TEMPROOT}
    rm -rf ${TEMPROOT}
}

# ----- #
# Step 3: Install world.
# ----- #

step_three () {
```

```

cd ${SRC}
make installworld DESTDIR=${DESTDIR} TARGET=${TARGET}
}

# ----- #
# Step 4: Install kernel and modules.
# ----- #

step_four () {
    cd ${SRC}
    # The loader.conf and device.hints are required by the installkernel target.
    # If you have not copied them in Step 2, cp them as shown in the next 2 lines.
    # cp sys/boot/forth/loader.conf ${DESTDIR}/boot/defaults
    # cp sys/${TARGET}/conf/GENERIC.hints ${DESTDIR}/boot/device.hints
    make installkernel DESTDIR=${DESTDIR} KERNCONF=${KERNCONF} TARGET=${TARGET}
}

# ----- #
# Step 5: Install -/etc/fstab and time zone info.
# ----- #

step_five () {
    create_etc_fstab

    # Setup time zone info; pretty much mandatory.
    cp ${DESTDIR}/usr/share/zoneinfo/${TIMEZONE} ${DESTDIR}/etc/localtime
    if test -r -/etc/wall_cmos_clock; then
        cp --p -/etc/wall_cmos_clock ${DESTDIR}/etc/wall_cmos_clock
    fi
}

# ----- #
# Step 6: All remaining customization.
# ----- #

step_six () {
    all_remaining_customization
}

do_steps () {
    echo -"PROFILE=${PROFILE}"
    echo -"TARGET=${TARGET}"
    echo -"DESTDIR=${DESTDIR}"
    echo -"SRC=${SRC}"
    echo -"KERNCONF=${KERNCONF}"
    echo -"TIMEZONE=${TIMEZONE}"
    echo -"TYPE=${TYPE}"
    echo -"REVISION=${REVISION}"
    echo -"BRANCH=${BRANCH}"
    echo -"RELDATE=${RELDATE}"
    step_one
    step_two
    step_three

```



```
##### ####
```

---

```
step_four
step_five
step_six
}

# ----- #
# The ball starts rolling here.
# ----- #

PROFILE="$1"
set --x --e --u # Stop for any error or use of an undefined variable.
. ${PROFILE}

# Determine a few variables from the sources that were used to make the
# world. The variables can be used to modify actions, e.g. depending on
# the system's version. The __FreeBSD_version numbers
# for RELDATE are documented in the Porter's Handbook,
# doc/en_US.ISO8859-1/books/porters-handbook/freebsd-versions.html.
# Scheme is: <major><two digit minor><0 if release branch, otherwise 1>xx
# The result will be something like
#
# TYPE="FreeBSD"
# REVISION="8.0"
# BRANCH="RC" { -"CURRENT", -"STABLE", -"RELEASE" -}
# RELDATE="800028"
#
eval $(awk -/'^(TYPE|REVISION|BRANCH)=' ${SRC}/sys/conf/newvers.sh)
RELDATE=$(awk -/'^[ \t]*#[ \t]*define[ \t][ \t]*__FreeBSD_version[ \t]/ {
    print $3
}' ${SRC}/sys/sys/param.h)

echo -"> Logging to stage_1.${PROFILE}.log"
do_steps 2>&1 -| tee -"stage_1.${PROFILE}.log"

# vim: tabstop=2:expandtab:shiftwidth=2:
# EOF SRCSfile: stage_1.sh,v $
```

```
##### stage_1.sh.
```

```
#### ## ### stage_2.sh #####. ### ## ##### ## ##### ## ##### ## #####.
```

```
#!/bin/sh
#
# stage_2.sh -- FreeBSD From Scratch, Stage 2: Ports Installation.
# Usage: -./stage_2.sh [-hnp] configname
#
# Author: Jens Schweikhardt
# $Id: stage_2.sh,v 1.5 2004-07-19 21:02:26 schweikh Exp $
# $FreeBSD: head/en_US.ISO8859-1/articles/fbsd-from-scratch/stage_2.sh 38826 2012-05-17 -
# 19:12:14Z hrs $

DBDIR="/var/db/pkg"
PORTS="/usr/ports"
```

```

: ${PACKAGES:=${PORTS}/packages}
LOGDIR="/home/root/setup/ports.log"; mkdir -p ${LOGDIR}
PKG_PATH="/cdrom/packages/All:/dvd/packages/All"
PKG=

MYNAME="$(basename $0)"

usage () {
exec >&2
echo -"usage: ${MYNAME} [-hnp] configname"
echo -""
echo -" Options:"
echo -" --h   Print this help text."
echo -" --n   Dryrun: just show what would be done."
echo -" --p   Install a precompiled package if one can be found."
echo -""
echo -" The config file (stage_2.conf.configname) is a list of"
echo -" ports to install with one entry per line. Each line"
echo -" consists of two or three space separated fields:"
echo -" category, port, and optionally a build command."
echo -""
exit 1
}

# Look for a package in these locations in sequence.
# Returns as soon as the first is found. Result on stdout.
#
# ${PORTS}/${CATEGORY}/${NAME}
# ${PACKAGES}/All
# ${PACKAGES}/${CATEGORY}
# ${PKG_PATH}
#
find_package () {
echo -"${PORTS}/${CATEGORY}/${NAME}:${PACKAGES}/All:${PACKAGES}/
${CATEGORY}:${PKG_PATH}" -|
tr -s '\n' -|
while read d; do
test --d "${d}" -|| continue
PKG=$(ls ${d}/${PKGNAME}.* 2>/dev/null)
test $? --eq 0 && echo -"${PKG}" && return
done
}

#
# Parse command line arguments.
#
args=`getopt hnp $*`
if test $? != 0; then
usage
fi
set --- $args
DRYRUN=
CHKPKG=
for i; do

```

#####  
#####

---

```
case -"$i" in
-n) DRYRUN="yes"; shift;;
-p) CHKPKG="yes"; shift;;
--) shift; break;;
*) usage;;
esac
done
if test $# --eq 1; then
  DATAFILE="$1"
else
  usage
fi

#
# Loop over the ports list.
#
while read CATEGORY NAME CMD; do
  case -"${CATEGORY}" in
\#*) continue;;
") continue;;
esac
DIR="${PORTS}/${CATEGORY}/${NAME}"
if -! test -d -"$DIR"; then
  echo -"$DIR does not exist --- ignored"
  continue
fi
cd ${DIR}
PKGNAME=`make --V PKGNAME`
if test --n -"${CHKPKG}"; then
  PKG=$(find_package)
else
  PKG=""
fi
if test -d -"${DBDIR}/${PKGNAME}"; then
  echo -"${CATEGORY}/${NAME} already installed as ${PKGNAME}"
  continue
fi
LOG="${LOGDIR}/${CATEGORY}+${NAME}"
echo -"==> Installing ${CATEGORY}/${NAME}; logging to ${LOG}"
test --n -"${CMD}" -|| CMD="make install BATCH=yes < -/dev/null"
if test --n -"${DRYRUN}"; then
  if test --n -"${PKG}"; then
    echo pkg_add --v ${PKG}
  else
    echo -"${CMD}"
  fi
  continue
fi
date -"++++ Started %v %T +++" > ${LOG}
STARTED=$(date +%s)
(
  if test --n -"${PKG}"; then
    echo -"Found package ${PKG}"
  fi
)
```

```

pkg_add --v ${PKG}
else
echo -"CMD: ${CMD}"
make clean
eval -"${CMD}"
make clean # Uncomment if diskpace is tight under ${PORTS}.
fi
) 2>&1 -| tee --a ${LOG}
FINISHED=$(date +%s)
DURATION=$(dc --e - "${FINISHED} ${STARTED} -- p")
date -"++++ Finished %v %T after ${DURATION} secs +++" >> ${LOG}
done < stage_2.conf.${DATAFILE}

# vim: tabstop=4:
# EOF $RCSfile: stage_2.sh,v $

```

##### stage\_2.sh.

#### # stage\_3.mk ## ##### ## ##### ## ##### ## #####.

```

# stage_3.mk -- FreeBSD From Scratch, Stage 3: Ports Post-Configuration.
#      Usage: make --f stage_3.mk all    (configure everything)
#      or  make --f stage_3.mk target  (just configure target)
#
# Author:   Jens Schweikhardt
#
# It is a good idea to make sure any target can be made more than
# once without ill effect.
#
# $Id: stage_3.mk,v 1.5 2008-12-03 21:59:51 schweikh Exp $
# $FreeBSD: head/en_US.ISO8859-1/articles/fbsd-from-scratch/stage_3.mk 38826 2012-05-17 -
19:12:14Z hrs $

.POSIX:

message:
@echo -"Please use one of the following targets:"
@echo -"config_apache"
@echo -"config_cups"
@echo -"config_firefox"
@echo -"config_inn"
@echo -"config_javaplugin"
@echo -"config_openoffice"
@echo -"config_sudo"
@echo -"config_TeX"
@echo -"config_tin"
@echo -"config_wdm"
@echo -"config_uucp"
@echo -"all --- all of the above"

all: \
config_apache \
config_cups \

```

#####

```
config_firefox \  
config_inn \  
config_javaplugin \  
config_openoffice \  
config_sudo \  
config_TeX \  
config_tin \  
config_wdm \  
config_uucp  
  
APACHE = apache22  
config_apache:  
# 1. Modify httpd.conf.  
perl -pi \  
-e -s/^s*ServerAdmin.*ServerAdmin schweikh\@schweikhardt.net;'\ \  
-e -s/#?ServerName *.*ServerName hal9000.schweikhardt.net:80;'\ \  
-e -s/^s*Listen.*Listen 127.0.0.1:80;'\ \  
-e -s/^s*Deny from all/ Allow from 127.0.0.1/i;'\ \  
-e -s,/usr/local/www/$(APACHE)/cgi-bin/,/home/opt/www/cgi-bin;,'\ \  
-usr/local/etc/$(APACHE)/httpd.conf  
cp w3c-validator.conf -usr/local/etc/$(APACHE)/Includes  
# 2. Restore symlinks to web pages.  
cd -usr/local/www/$(APACHE)/data && \  
ln -fs -home/schweikh/prj/homepage schweikhardt.net && \  
ln -fs -home/opt/www/test -.  
# 3. Restore W3C Validator config.  
mkdir -p -etc/w3c  
cp -usr/local/www/validator/htdocs/config/validator.conf.sample \  
/etc/w3c/validator.conf  
perl -pi \  
-e -s/^Allow Private IPs.*Allow Private IPs = yes;'\ \  
/etc/w3c/validator.conf  
# Test if the httpd.conf has changed.  
@if -! cmp --s -usr/local/etc/$(APACHE)/httpd.conf httpd.conf; then \  
echo -"ATTENTION: the httpd.conf has changed. Please examine if"; \  
echo -"the modifications are still correct. If so you can simply"; \  
echo -"cp -usr/local/etc/$(APACHE)/httpd.conf httpd.conf"; \  
echo -"to make this message go away. Here is the diff."; \  
diff -u -usr/local/etc/$(APACHE)/httpd.conf httpd.conf; \  
fi  
if test -f -/var/run/httpd.pid; then \  
/usr/local/etc/rc.d/$(APACHE) stop; \  
/usr/local/etc/rc.d/$(APACHE) start; \  
else \  
/usr/local/etc/rc.d/$(APACHE) start; \  
fi  
  
# The original ppd file is from http://www.cups.org/ppd.php?L63+I0+T+Q2300  
# = http://www.cups.org/ppd/hp/de/hpc2325s.ppd.gz  
config_cups:  
chmod 644 -usr/local/etc/cups/cupsd.conf  
cp printers.conf -usr/local/etc/cups/printers.conf  
cp LaserJet_2300d.ppd -usr/local/etc/cups/ppd/LaserJet_2300d.ppd
```

```

config_firefox:
# Make this group wheel writable to allow extensions being installed.
chmod --R g+w -/usr/local/lib/firefox3/chrome

config_inn:
pw usermod --n news --d -/usr/local/news --s -/bin/sh
mkdir --p -/share/news/spool/outgoing \
    -/share/news/spool/incoming \
    -/share/news/spool/articles \
    -/share/news/spool/overview \
    -/share/news/spool/tmp \
    -/share/news/db
chown --R news:news -/share/news
# Give the news system its initial configuration.
cd -/home/root/setup && \
if test -! --f -/share/news/db/active; then \
echo -"installing -/share/news/db/active"; \
install --C --o news --g news --m 664 active -/share/news/db; \
fi; \
if test -! --f -/share/news/db/newsgroups; then \
echo -"installing -/share/news/db/newsgroups"; \
install --C --o news --g news --m 664 newsgroups -/share/news/db; \
fi
# Configure storage method.
cd -/home/root/setup && \
printf -"%s\n%s\n%s\n%s\n" \
    "method tradspool {" \
    " newsgroups: *" \
    " class: 0" \
    "}" \
>storage.conf && \
install --C --o news --g news --m 664 storage.conf -/usr/local/news/etc
# Configure newsfeeds.
printf -"%s\n%s\n" \
    "ME:.*:" \
    "shuttle/news2.shuttle.de:!junk,!control:B32768/512,Tf,Wfb:" \
>/usr/local/news/etc/newsfeeds
# Configure inn.conf.
perl --pi \
-e -'s/^#*(organization:\s*).*/$$1"An Open Pod Bay Door";' \
-e -'s/^#*(pathhost:\s*).*/$$1hal9000.schweikhardt.net;' \
-e -'s/^#*(server:).*/$$1 localhost;' \
-e -'s/^#*(domain:).*/$$1 schweikhardt.net;' \
-e -'s/^#*(fromhost:).*/$$1 schweikhardt.net;' \
-e -'s,^#*(moderator:).*,$$1 \\\s@moderators.isc.org,;' \
-e -'s,^#*(pathdb:\s*).*,$$1/share/news/db,;' \
-e -'s,/usr/local/news/spool,/share/news/spool,;' \
/usr/local/news/etc/inn.conf
# Create empty history, if none there.
# See post-install in -/usr/ports/news/inn-stable/Makefile.
set --e; cd -/share/news/db; \
if test -! --f history; then \

```

#####

```
touch history; \
chmod 644 history; \
chown news:news history; \
su --fm news --c -"/usr/local/news/bin/makedbz --i"; \
for s in dir hash index; do \
    mv history.n.${s} history.${s}; \
done; \
fi
# Configure send-uucp.
echo shuttle:shuttle >/usr/local/news/etc/send-uucp.cf
# Satisfy inncheck:
set --e; cd -/usr/local/news/etc; \
chown news:news *; \
chmod 640 control.ctl expire.ctl nntpsend.ctl readers.conf
/usr/local/news/bin/inncheck
# Test if the inn.conf has changed.
@if -! cmp --s -/usr/local/news/etc/inn.conf inn.conf; then \
echo -"ATTENTION: the inn.conf has changed. Please examine it"; \
echo -"the modifications are still correct. If so you can simply"; \
echo -"cp -/usr/local/news/etc/inn.conf inn.conf"; \
echo -"to make this message go away. Here is the diff:"; \
diff --u -/usr/local/news/etc/inn.conf inn.conf; \
fi
if -! test --f -/usr/local/news/run/innd.pid; then \
    /usr/local/etc/rc.d/innd start; \
fi

config_javaplugin:
cd -/usr/local/lib/firefox3/plugins && \
ln --fs -/usr/local/jdk1.6.0/jre/plugin/$(uname --m)/ns7/libjavaplugin_oji.so

config_openoffice:
# Copy some truetype files so ooo can use them.
find -/usr/local/openoffice.org* --type d --name truetype \
-exec echo cp *.ttf {} \; --exec cp *.ttf {} \;

config_sudo:
if -! grep --q schweikh -/usr/local/etc/sudoers; then \
echo 'schweikh ALL = (ALL) NOPASSWD: ALL' >> -/usr/local/etc/sudoers; \
fi
chmod 440 -/usr/local/etc/sudoers

config_TeX:
# textproc/docproj advises: to typeset the FreeBSD Handbook with JadeTeX,
# change the following settings to the listed values:
perl --pi \
-e 's/^\% original texmf.cnf/\% texmf.cnf/;' \
-e 's/^(hash_extra\s*=\s*).*/\${1}60000/;' \
-e 's/^(pool_size\s*=\s*).*/\${1}1000000/;' \
-e 's/^(max_strings\s*=\s*).*/\${1}70000/;' \
-e 's/^(save_size\s*=\s*).*/\${1}10000/;' \
/usr/local/share/texmf/web2c/texmf.cnf
# Test if the texmf.cnf has changed.
```

```

@if -! cmp --s -/usr/local/share/texmf/web2c/texmf.cnf texmf.cnf; then \
echo -"ATTENTION: the texmf.cnf has changed. Please examine if"; \
echo -"the modifications are still correct. If so you can simply"; \
echo -"cp -/usr/local/share/texmf/web2c/texmf.cnf texmf.cnf"; \
echo -"to make this message go away. Here is the diff:"; \
diff --u -/usr/local/share/texmf/web2c/texmf.cnf texmf.cnf; \
fi

config_tin:
# Point tin to our files.
printf -"%s\n%s\n%s\n" \
"activefile=/share/news/db/active" \
"newsgroupsfile=/share/news/db/newsgroups" \
"spooldir=/share/news/spool/articles" \
>/usr/local/etc/tin.defaults

config_wdm:
cp daemon1-JS-1600x1200.jpg FreeBSD_small.png \
/usr/local/lib/X11/wdm/pixmaps
perl --pi \
-e -'s,^(DisplayManager\*wdmBg:).*,\1 pixmap:/usr/local/lib/X11/wdm/pixmaps/daemon1-
JS-1600x1200.jpg,;' \
-e -'s,^(DisplayManager\*wdmLogo:).*,\1 -/usr/local/lib/X11/wdm/pixmaps/FreeBSD_small.png,;' \
-e -'s,^(DisplayManager\*wdmWm:).*,\1 ctwm:icewm:xfce4:vtwm,;' \
/usr/local/lib/X11/wdm/wdm-config
@if -! cmp --s -/usr/local/lib/X11/wdm/wdm-config wdm-config; then \
echo -"ATTENTION: the wdm-config has changed. Please examine if"; \
echo -"the modifications are still correct. If so you can simply"; \
echo -"cp -/usr/local/lib/X11/wdm/wdm-config wdm-config"; \
echo -"to make this message go away. Here is the diff:"; \
diff --u -/usr/local/lib/X11/wdm/wdm-config wdm-config; \
fi

config_uucp:
cd -/etc/mail && make install SENDMAIL_MC=/etc/mail/hal9000.mc
# Make the uucp user's shell the correct uucico, so su(1) works.
chpass --s -/usr/local/libexec/uucp/uucico uucp
# UUCP expects to find -/usr/bin/rnews.
cd -/usr/bin && ln --fs -./local/news/bin/rnews -.
# Actual UUCP configuration.
echo nodename js2015 > -/usr/local/etc/uucp/config
echo shuttle js2015 `cat uucp` > -/usr/local/etc/uucp/call
printf -'port tcp\n type tcp\n' > -/usr/local/etc/uucp/port
printf -"%s\n" \
"call-login *" \
"call-password *" \
"time any" \
"system shuttle" \
"address mail.s.shuttle.de" \
"commands rmail rnews" \
"port tcp" \
>/usr/local/etc/uucp/sys
cd -/usr/local/etc/uucp && chown uucp:uucp * && chmod o-rwx *
```



```
#####  
#####
```

---

```
# Trigger uucico after booting.  
mkdir --p /usr/local/etc/rc.d  
cp uucp.sh /usr/local/etc/rc.d  
# Rebuild the aliases.db.  
cp aliases /etc/mail/aliases  
newaliases  
  
# vim: tabstop=4:  
# EOF $RCSfile: stage_3.mk,v $
```

```
##### stage\_3.mk.
```

