$SPAD/src Makefile

Timothy Daly

December 3, 2016

Abstract

Contents

1 Directory overview

1.1 Environment variables

DIRS is a list of directories with the suffix **dir**.

There is a partial order of the directories which is not apparent.

The bootdir must occur first as it builds the boot language compiler and the bootsys image.

The interpdir must come second because it builds the interpreter and the interpsys image, which is the basic algebra engine.

The sharedir must occur before the algebradir because it needs to put the databases in the proper location.

The algebradir comes next. It contains the source code for all of the algebra that Axiom knows.

The etcdir must occur after the algebradir because it builds the databases from all of the *.nrlib directories constructed by algebradir.

The inputdir needs to occur last because it tests various other parts of the system.

The SRCDIRS variable is a list of directory targets. Since this varies from system to system it has been lifted up to the top level Makefile.pamphlet. If you add a new stanza to this Makefile you should check that list.

— **environment** —

```
SETUP=scriptsdir libdir
DIRS=${SRCDIRS}
LTANGLE=${OBJ}/${SYS}/bin/lisp
VOL11=${BOOKS}/bookvol11.pamphlet
```

————-

1.2 The scripts directory

The **scripts** directory contains shell scripts that we use to simplify system builds. They are generally copied directly into the final executable path. Perhaps this design should be reconsidered if we don't plan to ship them in the final system.

— **scriptsdir** —

```
scriptsdir: ${SRC}/scripts/Makefile
@echo 1 making ${SRC}/scripts
@(cd scripts ; ${ENV} ${MAKE} )

${SRC}/scripts/Makefile: ${SRC}/scripts/Makefile.pamphlet
@echo 2 making ${SRC}/scripts/Makefile from \
           ${SRC}/scripts/Makefile.pamphlet
@( cd scripts ; \
           ${EXTRACT} Makefile ; \
```

```
           cp Makefile.pdf ${MNT}/${SYS}/doc/src/scripts.Makefile.pdf )

scriptsdocument: ${SRC}/scripts/Makefile
@echo 3 documenting ${SRC}/scripts
@mkdir -p ${INT}/doc/src/scripts
@( cd scripts ; ${ENV} ${MAKE} document )

scriptsclean: ${SRC}/scripts/Makefile
@echo 4 cleaning ${SRC}/scripts
@( cd scripts ; ${ENV} ${MAKE} clean )
@rm -f ${SRC}/scripts/Makefile
@rm -f ${SRC}/scripts/Makefile.dvi
```

———-

1.3 The clef directory

The **clef** directory contains an Axiom command that works similar to GNU Readline.

— **clefdir** —

```
clefdir: ${SRC}/clef/Makefile
@echo 5 making ${SRC}/clef
@mkdir -p ${OBJ}/${SYS}/clef
@mkdir -p ${MNT}/${SYS}/doc/src/clef
@(cd clef ; ${ENV} ${MAKE} )

${SRC}/clef/Makefile: ${SRC}/clef/Makefile.pamphlet
@echo 6 making ${SRC}/clef/Makefile from ${SRC}/clef/Makefile.pamphlet
@( cd clef ; \
           ${EXTRACT} Makefile ; \
           cp Makefile.pdf ${MNT}/${SYS}/doc/src/clef.Makefile.pdf )

clefdocument: ${SRC}/clef/Makefile
@echo 7 documenting ${SRC}/clef
@mkdir -p ${INT}/doc/src/clef
@( cd clef ; ${ENV} ${MAKE} document )

clefclean: ${SRC}/clef/Makefile
@echo 8 cleaning ${SRC}/clef
@( cd clef ; ${ENV} ${MAKE} clean )
@rm -f ${SRC}/clef/Makefile
@rm -f ${SRC}/clef/Makefile.dvi
```

———-

1.4 The clef directory

Superman (sman) is the master process that runs all of the other processes including axiom, clef, nagman, graphics, and hyperdoc

— **smandir** —

```
smandir: ${SPD}/books/bookvol6.pamphlet
@echo 5 making ${SRC}/sman
@mkdir -p ${INT}/sman
@mkdir -p ${OBJ}/${SYS}/sman
@mkdir -p ${MNT}/${SYS}/doc/src/sman
@( cd sman ; \
          echo '(tangle "${SPD}/books/bookvol6.pamphlet" "*" "Makefile")' \
        | ${LTANGLE} ; \
           ${ENV} ${MAKE} )
```

————-

1.5 The share directory

The **share** directory files that are shared by all version of the system.

— **sharedir** —

```
sharedir: ${SRC}/share/Makefile
@echo 9 making ${SRC}/share
@mkdir -p ${MNT}/${SYS}/lib
@(cd share ; ${ENV} ${MAKE} )

${SRC}/share/Makefile: ${SRC}/share/Makefile.pamphlet
@echo 10 making ${SRC}/share/Makefile from \
           ${SRC}/share/Makefile.pamphlet
@( cd share ; \
           ${EXTRACT} Makefile ; \
           cp Makefile.pdf ${MNT}/${SYS}/doc/src/share.Makefile.pdf )

sharedocument: ${SRC}/share/Makefile
@echo 11 documenting ${SRC}/share
@mkdir -p ${INT}/doc/src/share
@( cd share ; ${ENV} ${MAKE} document )

shareclean: ${SRC}/share/Makefile
@echo 12 cleaning ${SRC}/share
@( cd share ; ${ENV} ${MAKE} clean )
@rm -f ${SRC}/share/Makefile
@rm -f ${SRC}/share/Makefile.dvi
```

————-

1.6 The booklet directory

The **booklet** directory contains pamphlet files that document Axiom at a "higher level" than any particular pamphlet file. Booklets can be stand-alone descriptions (e.g. the Rosetta.pamphlet), top-down slices thru the system (e.g. the Integration.pamphlet), or horizontal slices thru the system (e.g. the Matrix.pamphlet).

— **bookletsdir** —

```
bookletsdir: ${SRC}/booklets/Makefile
@echo 13 making ${SRC}/booklets
@(cd booklets ; ${ENV} ${MAKE} )

${SRC}/booklets/Makefile: ${SRC}/booklets/Makefile.pamphlet
@echo 14 making ${SRC}/booklets/Makefile from \
          ${SRC}/booklets/Makefile.pamphlet
@( cd booklets ; ${DOCUMENT} ${NOISE} Makefile )

bookletsdocument: ${SRC}/booklets/Makefile
@echo 15 documenting ${SRC}/booklets
@mkdir -p ${INT}/doc/src/booklets
@( cd booklets ; ${ENV} ${MAKE} )

bookletsclean: ${SRC}/booklets/Makefile
@echo 16 cleaning ${SRC}/booklets
@( cd booklets ; ${ENV} ${MAKE} clean )
```

———-

1.7 The lib directory

The **lib** directory is used to build **libspad.a** which contains C code for extending the underlying Common Lisp systems. It is built early in the process of system building because we need to make **libspad.a** before we make the Common Lisps.

— **libdir** —

```
libdir: ${SRC}/lib/Makefile
@echo 17 making ${SRC}/lib
@mkdir -p ${INT}/lib
@mkdir -p ${OBJ}/${SYS}/lib
@mkdir -p ${INT}/doc/src/lib
@mkdir -p ${MNT}/${SYS}/doc/src/lib
@(cd lib ; ${ENV} ${MAKE} )

${SRC}/lib/Makefile: ${SRC}/lib/Makefile.pamphlet
@echo 18 making ${SRC}/lib/Makefile from ${SRC}/lib/Makefile.pamphlet
@( cd lib ; \
```

```
           ${EXTRACT} Makefile ; \
           cp Makefile.pdf ${MNT}/${SYS}/doc/src/lib.Makefile.pdf )

libdocument: ${SRC}/lib/Makefile
@echo 19 documenting ${SRC}/lib
@mkdir -p ${INT}/doc/src/lib
@( cd lib ; ${ENV} ${MAKE} document )

libclean: ${SRC}/lib/Makefile
@echo 20 cleaning ${SRC}/lib
@( cd lib ; ${ENV} ${MAKE} clean )
@rm -rf ${OBJ}/${SYS}/lib
@rm -f ${SRC}/lib/Makefile
@rm -f ${SRC}/lib/Makefile.dvi
```

———-

1.8 The boot directory

Axiom is built in layers. The first layer is contructed into an image called **bootsys**. The **bootsys** image is used to translate boot code to common lisp code. Since a boot coded interpreter is needed to translate the code for the boot coded interpreter we have a "boot-strapping" problem. In order to get the whole process to start we need certain files kept in common lisp form. This directory contains those files.

— **bootdir** —

```
bootdir: ${SRC}/boot/Makefile
@echo 21 making ${SRC}/boot
@mkdir -p ${INT}/boot
@mkdir -p ${OBJ}/${SYS}/boot
@mkdir -p ${MNT}/${SYS}/doc/src/boot
@(cd boot ; ${ENV} ${MAKE} )

${SRC}/boot/Makefile: ${SRC}/boot/Makefile.pamphlet
@echo 22 making ${SRC}/boot/Makefile from ${SRC}/boot/Makefile.pamphlet
@( cd boot ; \
           ${EXTRACT} Makefile ; \
           cp Makefile.pdf ${MNT}/${SYS}/doc/src/boot.Makefile.pdf )

bootdocument: ${SRC}/boot/Makefile
@echo 23 documenting ${SRC}/boot
@mkdir -p ${MNT}/${SYS}/doc/src/boot
@mkdir -p ${INT}/doc/src/boot
@( cd boot ; ${ENV} ${MAKE} document )

bootclean: ${SRC}/boot/Makefile
@echo 24 cleaning ${SRC}/boot
```

```
@( cd boot ; ${ENV} ${MAKE} clean )
@rm -rf ${OBJ}/${SYS}/boot
@rm -f ${SRC}/boot/Makefile
@rm -f ${SRC}/boot/Makefile.dvi
```

———-

1.9 The interp directory

Once **bootsys** exists we need to build **depsys** and **interpsys**. Since these two images share a lot of files they are built in the interp subdirectory using the same Makefile.

Book Volume 5 contains the interpreter.

Book Volume 9 contains the compiler.

We need the file util.ht from the hyperdoc pages (bookvol7.1) because the spad compiler checks the syntax of the ++ comments against the macros that exist in that file. This is done in the function buildHtMacroTable (htcheck.boot)

— **interpdir** —

```
interpdir: ${SRC}/interp/Makefile
@echo 25 making ${SRC}/interp
@mkdir -p ${INT}/interp
@mkdir -p ${INT}/algebra
@mkdir -p ${OBJ}/${SYS}/interp
@mkdir -p ${MNT}/${SYS}/autoload
@mkdir -p ${MNT}/${SYS}/algebra
@mkdir -p ${MNT}/${SYS}/doc/msgs
@mkdir -p ${MNT}/${SYS}/doc/src/interp
@cp ${SPD}/books/bookvol5.pamphlet interp
@cp ${SPD}/books/bookvol9.pamphlet interp
@cp ${SPD}/books/bookvol10.5.pamphlet interp
@(cd ${MNT}/${SYS}/doc ; \
      echo '(tangle "${SPD}/books/bookvol7.1.pamphlet" "util.ht" "util.ht")' \
       | ${LTANGLE} )
@(cd interp ; ${ENV} ${MAKE} )

${SRC}/interp/Makefile: ${SRC}/interp/Makefile.pamphlet
@echo 26 making ${SRC}/interp/Makefile from \
                 ${SRC}/interp/Makefile.pamphlet
@( cd interp ; \
           ${EXTRACT} Makefile ; \
           cp Makefile.pdf ${MNT}/${SYS}/doc/src/interp.Makefile.pdf )

interpdocument: ${SRC}/interp/Makefile
@echo 27 documenting ${SRC}/interp
@mkdir -p ${MNT}/${SYS}/doc/src/interp
@mkdir -p ${INT}/doc/src/interp
```

```
@( cd interp ; ${ENV} ${MAKE} document )

interpclean: ${SRC}/interp/Makefile
@echo 28 cleaning ${SRC}/interp
@( cd interp ; ${ENV} ${MAKE} clean )
@rm -rf ${OBJ}/${SYS}/interp
@rm -f ${SRC}/interp/Makefile
@rm -f ${SRC}/interp/Makefile.dvi
```

———-

1.10 The input directory

The input directory contains code used for examples, regression testing, and bug tracking. In a shipped system the working examples are collected and documented so a user can learn how to use Axiom's many domains and packages.

During development the input files can be used for tracking bugs and testing fixes of known bugs. Once a bug is fixed it is moved to the regression test set.

Prior to building a shippable system all of the input files are run with the example code and the regression test code. Regression test input files are compared against known good results to ensure that nothing has been broken in the process of fixing bugs.

— **inputdir** —

```
inputdir: ${SRC}/input/Makefile
@echo 33 making ${SRC}/input
@mkdir -p ${INT}/input
@mkdir -p ${MNT}/${SYS}/input
@mkdir -p ${MNT}/${SYS}/doc/src/input
@(cd input ; ${ENV} ${MAKE} )
# @(cd input ; ${ENV} ${MAKE} -j 10)

${SRC}/input/Makefile: ${SRC}/input/Makefile.pamphlet
@echo 34 making ${SRC}/input/Makefile from \
                ${SRC}/input/Makefile.pamphlet
@( cd input ; \
           ${EXTRACT} Makefile ; \
           cp Makefile.pdf ${MNT}/${SYS}/doc/src/input.Makefile.pdf )

inputdocument: ${SRC}/input/Makefile
@echo 35 documenting ${SRC}/input
@mkdir -p ${INT}/doc/src/input
@( cd input ; ${ENV} ${MAKE} document )

inputclean: ${SRC}/input/Makefile
@echo 36 cleaning ${SRC}/input
@( cd input ; ${ENV} ${MAKE} clean )
```

```
@rm -rf ${OBJ}/${SYS}/input
@rm -f ${SRC}/input/Makefile
@rm -f ${SRC}/input/Makefile.dvi
```

———-

1.11 The etc directory

The etc directory contains code used as tools surrounding Axiom.

The asq [2] command, contained in this directory, is useful for finding detailed information about domains, packages, and categories from the shell without running Axiom.

— **etcdir** —

```
etcdir: ${SRC}/etc/Makefile
@echo 37 making ${SRC}/etc
@mkdir -p ${OBJ}/${SYS}/etc
@mkdir -p ${MNT}/${SYS}/bin
@mkdir -p ${MNT}/${SYS}/lib
@(cd etc ; ${ENV} ${MAKE} )

${SRC}/etc/Makefile: ${SRC}/etc/Makefile.pamphlet
@echo 38 making ${SRC}/etc/Makefile from ${SRC}/etc/Makefile.pamphlet
@( cd etc ; \
           ${EXTRACT} Makefile ; \
           cp Makefile.pdf ${MNT}/${SYS}/doc/src/etc.Makefile.pdf )

etcdocument: ${SRC}/etc/Makefile
@echo 39 documenting ${SRC}/etc
@mkdir -p ${INT}/doc/src/etc
@( cd etc ; ${ENV} ${MAKE} document )

etcclean: ${SRC}/etc/Makefile
@echo 40 cleaning ${SRC}/etc
@( cd etc ; ${ENV} ${MAKE} clean )
@rm -rf ${OBJ}/${SYS}/etc
@rm -f ${SRC}/etc/Makefile
@rm -f ${SRC}/etc/Makefile.dvi
```

———-

1.12 The doc directory

The doc directory contains code used for documenting Axiom.

We've added the books directory above as part of the literate process. This takes all of the Axiom volumes and creates .pdf copies in the final doc directory.

— **docdir** —

```
docdir: ${SRC}/doc/Makefile
@echo 41 making ${SRC}/doc
@mkdir -p ${INT}/doc
@mkdir -p ${MNT}/${SYS}/bin
@mkdir -p ${MNT}/${SYS}/doc/hypertex/bitmaps
@(cd ${MNT}/${SYS}/doc/hypertex ; \
         echo '(tangle "${VOL11}" "axiom1.bitmap" "bitmaps/axiom1.bitmap")' \
               | ${LTANGLE} ; \
         echo '(tangle "${VOL11}" "rcm3720.input" "rcm3720.input")' \
               | ${LTANGLE} ; \
 echo '(tangle "${VOL11}" "strang.input" "strang.input")' \
               | ${LTANGLE} ; \
 echo '(tangle "${VOL11}" "signatures.txt" "signatures.txt")' \
               | ${LTANGLE} )
@ cp ${BOOKS}/ps/doctitle.png ${AXIOM}/doc/hypertex/doctitle.png
@ cp ${BOOKS}/ps/lightbayou.png ${AXIOM}/doc/hypertex/lightbayou.png
@(cd ${MNT}/${SYS}/doc/hypertex ; \
  ${BOOKS}/tanglec ${SPD}/books/bookvol11.pamphlet >Makefile11 ; \
  ${ENV} ${MAKE} -f Makefile11 ; \
  rm -f Makefile11 )
@(cd doc ; ${ENV} ${MAKE} )

${SRC}/doc/Makefile: ${SRC}/doc/Makefile.pamphlet
@echo 42 making ${SRC}/doc/Makefile from ${SRC}/doc/Makefile.pamphlet
@( cd doc ; \
           ${EXTRACT} Makefile ; \
           cp Makefile.pdf ${MNT}/${SYS}/doc/src/doc.Makefile.pdf )

docdocument: ${SRC}/doc/Makefile
@echo 43 documenting ${SRC}/doc
@mkdir -p ${INT}/doc/src/doc
@( cd doc ; ${ENV} ${MAKE} document )

docclean: ${SRC}/doc/Makefile
@echo 44 cleaning ${SRC}/doc
@( cd doc ; ${ENV} ${MAKE} clean )
@rm -rf ${OBJ}/${SYS}/doc
@rm -f ${SRC}/doc/Makefile
@rm -f ${SRC}/doc/Makefile.dvi
```

———-

1.13 Volume 7:Axiom Hyperdoc book

Hyperdoc is the Axiom document browser.

— **hyperdir** —

```
hyperdir: ${SPD}/books/bookvol7.pamphlet ${SPD}/books/bookvol7.1.pamphlet
@echo 7 making hyperdoc from bookvol7
```

```
@mkdir -p ${INT}/hyper
@mkdir -p ${OBJ}/${SYS}/hyper
@mkdir -p ${OBJ}/${SYS}/bin
@mkdir -p ${OBJ}/${SYS}/lib
@mkdir -p ${MNT}/${SYS}/doc/src/hyper
@(cd ${INT}/hyper ; \
          echo '(tangle "${SPD}/books/bookvol7.pamphlet" "*" "Makefile")' \
          | ${LTANGLE} ; \
  ${ENV} ${MAKE} )
@echo 7.1 making pages from bookvol7.1 ${ENV}
@(cd ${INT}/hyper ; \
          echo '(tangle "${SPD}/books/bookvol7.1.pamphlet" "*" "Makefile")' \
          | ${LTANGLE} ; \
  ${ENV} ${MAKE} )
```

———-

1.14 Volume 8: Axiom Graphics book

— **graphdir** —

```
graphdir: ${SPD}/books/bookvol8.pamphlet
@echo 8 making graph from bookvol8
@mkdir -p ${OBJ}/${SYS}/graph
@(cd ${OBJ}/${SYS}/graph ; \
 echo '(tangle "${SPD}/books/bookvol8.pamphlet" "*" "Makefile")' \
               | ${LTANGLE} ; \
  ${ENV} ${MAKE} )
```

———-

1.15 The algebra directory

The algebra directory contains code written in Axiom's computer algebra language called **spad**. There are two compilers for this language, the spad compiler and the **Aldor**[1] compiler.

Both of these compilers accept the same input language except for some platform-dependent differences. The spad compiler is written in Common Lisp (well, in boot, anyway) and is built into the interpreter. The Aldor compiler is written in C and runs stand-alone. Both compile files that will run in Axiom's interpreter. Files which end in ".spad" use the internal spad compiler. Files which end in ".as" use the external Aldor compiler.

1.15.1 Making the Makefile

The main body of the algebra Makefile is extracted from the Makefile.pamphlet file as usual. It contains generic rules for making all the .spad files in a series

of "layers" such that each layer depends on only those layers that preceed it, beginning with the bootstap layer. Because the individual .spad files are grouped into higher-level algebra pamphlet files, the rules for extracting them are derived from a simple script findAlgebraFiles which appends these additional rules to the Makefile.

The src/algebra/Makefile is specially constructed in two steps. The first step uses the tangle command to extract the normal Makefile information.

The second step is to extend the src/algebra/Makefile with stanzas that describe the steps to extract the algebra from the src/algebra/*.pamphlet files into the int/algebra/*.spad files. Further details are provided in Makefile for src/algebra.

The doc/spadhelp directory contains flat files of help text for the help system command. Algebra pamphlet contain examples that can be shown from these commands.

We need to make the int/input file here because the algebra Makefile will extract input files for regression testing from the algebra pamphlets.

We copy bookvol10.1 (Categories), bookvol10.2 (Domains), and bookvol10.3 (Packages) to the src/algebra subdirectory.

1.16 Volume 10: Axiom Algebra book

— algebradir —

```
algebradir: ${SRC}/algebra/Makefile ${SPD}/books/bookvol10.pamphlet \
            ${SPD}/books/bookvol7.1.pamphlet
@echo 29 making ${SRC}/algebra
@mkdir -p ${INT}/algebra
@mkdir -p ${INT}/input
@mkdir -p ${OBJ}/${SYS}/algebra
@mkdir -p ${MNT}/${SYS}/algebra
@mkdir -p ${MNT}/${SYS}/doc/src/algebra
@mkdir -p ${MNT}/${SYS}/doc/spadhelp
@mkdir -p ${MNT}/${SYS}/src/algebra
@mkdir -p ${INT}/input
@(cd algebra ; ${ENV} ${MAKE} )

${SRC}/algebra/Makefile: ${BOOKS}/bookvol10.pamphlet
@echo 30 making ${SRC}/algebra/Makefile from \
                ${BOOKS}/bookvol10.pamphlet
@( cd algebra ; \
${BOOKS}/tanglec ${BOOKS}/bookvol10.pamphlet Makefile >Makefile ; \
cp ${SPD}/books/bookvol10.2.pamphlet . ; \
cp ${SPD}/books/bookvol10.3.pamphlet . ; \
cp ${SPD}/books/bookvol10.4.pamphlet . ; \
cp ${SPD}/books/bookvol10.5.pamphlet . ; \
echo 30a tangling findAlgebraFiles from \
                ${BOOKS}/bookvol10.pamphlet ; \
```

```
${BOOKS}/tanglec ${BOOKS}/bookvol10.pamphlet \
             findAlgebraFiles >${INT}/algebra/findAlgebraFiles ; \
echo 30b running ${INT}/algebra/findAlgebraFiles ; \
. ${INT}/algebra/findAlgebraFiles >> Makefile )

algebraclean: ${SRC}/algebra/Makefile
@echo 32 cleaning ${SRC}/algebra
@( cd algebra ; ${ENV} ${MAKE} clean )
@rm -rf ${OBJ}/${SYS}/algebra
@rm -f ${SRC}/algebra/axiom.sty
@rm -f ${SRC}/algebra/*.spad
@rm -f ${SRC}/algebra/bookvol*
@rm -f ${SRC}/algebra/Makefile
@rm -f ${SRC}/algebra/Makefile.dvi
```

———-

1.17 Volume 11: Axiom Browser book

These are the pages for the Axiom Firefox browser interface. They can be expanded in parallel since they (currently) need no post processing. Thus the "-j 10" argument to make. We use "Makefile11" so that future parallel builds don't step on each other.

— **browserdir** —

```
browserdir: ${SPD}/books/bookvol11.pamphlet
@echo 11 making browser from bookvol11
# @mkdir -p ${MNT}/${SYS}/doc/hypertex/bitmaps
# @(cd ${MNT}/${SYS}/doc/hypertex ; \
#   ${BOOKS}/tanglec ${SPD}/books/bookvol11.pamphlet >Makefile11 ; \
#   ${ENV} ${MAKE} -j 10 -f Makefile11 ; \
#   rm -f Makefile11 )
```

———-

2 The Makefile

This Makefile gets called twice during the system build. The first call is to the **setup** stanza which will execute the **scriptsdir** stanza to copy the system scripts to the `mnt/${SYS}/bin` directory. And the **libdir** stanza is executed to build **libspad.a** which contains code needed by the underlying lisp.

The second call will execute the **all** stanza. This stanza walks all of the lower level directories.

— * —

```
\getchunk{environment}
```

```
all: announce ${DIRS}
@echo 49 finished ${SRC}

announce:
@ echo ====================================
@ echo src BUILDING SRC
@ echo ====================================

setup: ${SETUP}

\getchunk{scriptsdir}
\getchunk{clefdir}
\getchunk{smandir}
\getchunk{sharedir}
\getchunk{docdir}
\getchunk{bookletsdir}
\getchunk{libdir}
\getchunk{bootdir}
\getchunk{interpdir}
\getchunk{algebradir}
\getchunk{inputdir}
\getchunk{etcdir}

\getchunk{hyperdir}
\getchunk{graphdir}
\getchunk{browserdir}
```

———-

References

[1] Watt, Stephen, The Aldor compiler, **www.aldor.org**

[2] $AXIOM/src/etc/asq.c.pamphlet

[3] $AXIOM/src/clef/edible.c.pamphlet