

Problem

How can I build/test only SPECIFIC modules in the ACS subsystem Makefile?

Solution

In order to build/test only some ACS modules you have to overwrite the MODULES make variable while invoking the required target.

For example, if you want to rebuild only the modules recovery, baci, enumprop and maci just issue the following command:

```
make MODULES="LGPL/CommonSoftware/recovery LGPL/CommonSoftware/baci LGPL/CommonSoftware/enumprop LGPL/CommonSoftware/mac" rebuild
```

To see all modules that would be handled by a target in the ACS subsystem Makefile, use the show_modules target.

For example:

```
[almamgr@te20 ACS]$ make show_modules
Modules in build list are:
LGPL/Kit/vlt LGPL/Kit/doc LGPL/Kit/acs LGPL/Kit/acstempl LGPL/Tools/cmm LGPL/Tools/emacs LGPL/Tools/compat
LGPL/Tools/doxygen LGPL/Tools/tat LGPL/Tools/expat LGPL/Tools/loki LGPL/Tools/extjars LGPL/Tools/extpy LGPL
/CommonSoftware/acsutil LGPL/CommonSoftware/jacsutil LGPL/CommonSoftware/logging LGPL/CommonSoftware/acserr LGPL
/CommonSoftware/acsccomponent LGPL/CommonSoftware/cdb LGPL/CommonSoftware/cdbBrowser LGPL/CommonSoftware
/recovery LGPL/CommonSoftware/baci LGPL/CommonSoftware/enumprop LGPL/CommonSoftware/mac LGPL/CommonSoftware
/jacscommon LGPL/CommonSoftware/jmanager LGPL/CommonSoftware/acsd LGPL/CommonSoftware/acstime LGPL
/CommonSoftware/acsn LGPL/CommonSoftware/acslg LGPL/CommonSoftware/acssamp LGPL/CommonSoftware/abeans LGPL
/CommonSoftware/abeansgen LGPL/CommonSoftware/acsexmpl LGPL/CommonSoftware/acsabeans LGPL/CommonSoftware/objexp
LGPL/CommonSoftware/admind LGPL/CommonSoftware/jlog LGPL/CommonSoftware/acspy LGPL/CommonSoftware/xmljbind LGPL
/CommonSoftware/comphelpgen LGPL/CommonSoftware/XmlIdl LGPL/CommonSoftware/define LGPL/CommonSoftware
/acstestentities LGPL/CommonSoftware/acsjlog LGPL/CommonSoftware/jcont LGPL/CommonSoftware/jcontnc LGPL
/CommonSoftware/jcontexmpl LGPL/acsBUILD NO-LGPL/sla NO-LGPL/cfitsio
```

It is usually convenient to do:

```
> make show_modules
```

cat the modules you are interested in and paste them in the new make command overwriting MODULES.

For example, if you want to test (or continue and interrupted test) all modules with the exception of the Kit and Tools subsets, just issue:

```
> make MODULES='LGPL/CommonSoftware/acsutil LGPL/CommonSoftware/jacsutil LGPL/CommonSoftware/logging LGPL
/CommonSoftware/acserr LGPL/CommonSoftware/acsccomponent LGPL/CommonSoftware/cdb LGPL/CommonSoftware/cdbBrowser
LGPL/CommonSoftware/recovery LGPL/CommonSoftware/baci LGPL/CommonSoftware/enumprop LGPL/CommonSoftware/mac LGPL
/CommonSoftware/jacscommon LGPL/CommonSoftware/jmanager LGPL/CommonSoftware/acsd LGPL/CommonSoftware/acstime
LGPL/CommonSoftware/acsn LGPL/CommonSoftware/acslg LGPL/CommonSoftware/acssamp LGPL/CommonSoftware/abeans LGPL
/CommonSoftware/abeansgen LGPL/CommonSoftware/acsexmpl LGPL/CommonSoftware/acsabeans LGPL/CommonSoftware/objexp
LGPL/CommonSoftware/admind LGPL/CommonSoftware/jlog LGPL/CommonSoftware/acspy LGPL/CommonSoftware/xmljbind LGPL
/CommonSoftware/comphelpgen LGPL/CommonSoftware/XmlIdl LGPL/CommonSoftware/define LGPL/CommonSoftware
/acstestentities LGPL/CommonSoftware/acsjlog LGPL/CommonSoftware/jcont LGPL/CommonSoftware/jcontnc LGPL
/CommonSoftware/jcontexmpl LGPL/acsBUILD" test
```

Related articles

- [How can more people do development with ACS on the same machine without disturbing each other?](#)
- [Which ports are used by ACS?](#)
- [Problems connecting to ACS servers on a remote machine: bad /etc/hosts](#)
- [Why does the getComponent method of ZLegacy/ACS.ContainerServices return an object of type None?](#)
- [Why are some of my print statements not showing up in the container output section of acscommandcenter?](#)