

Problem

Is it mandatory that I install ACS in the /alma directory?

Solution

Several people have wondered whether ACS requires installation in /alma or whether another directory name can be used. For example, you may have noticed (in the .acs/.bash_profile.acs file) an environment variable named ALMASW_ROOTDIR and wondered whether it could be as simple as changing the installation directory (to something other than /alma) and updating the .bash_profile.acs to define ALMASW_ROOTDIR to point to the new directory (rather than to /alma). Unfortunately, though this seems logical, it won't work.

Simply changing the installation location of ACS and updating the ALMASW_ROOTDIR environment variable will **NOT** work, because Python has some hardcoded absolute paths that are *generated at build time* and therefore it is not "relocatable". But there are a number of things you can do instead, depending on what your goal/problem is:

A. If you really need to install ACS somewhere else, for example because your sysadmin will not give you access to the /alma directory, you can rebuild the tools in the location where you will be installing ACS; by rebuilding there, the python paths which are generated at build time will be correct for your installation:

- Change ALMASW_ROOTDIR and logout/login
- Get ACS/ExtProd from CVS, with the right tag.
- cd ACS/ExtProd/INSTALL
- make all
- then you can either rebuild ACS the usual way or unpack ACSSW and acsdata (in the new directory) from a binary distribution (note: in case it isn't clear why unpacking is ok; unpacking ACSSW into /YOUR_INSTALLATION_DIR/ACS-5.0/ works because the hard-coded paths generated during the build for python code will be constructed properly when you rebuild ACS/ExtProd from the new location. The code under ACSSW does not contain any python paths that must be constructed in this manner, so it is safe to simply unpack it without a rebuild. In other words, all python paths that must be generated at build time reside under the ACS/ExtProd directory).

B. If you can use /alma but you want to put an ACS-5.0 complete tree somewhere else, you can get away with symbolic links; as long as Python can find its directory in /alma/ACS-5.0 it will work.

C. Normally you should not need to have multiple installations of the tools, but only of ACSSW, for the different patches. In that case you can have multiple ACSSW-X.Y.Z directories inside ACS-5.0 and change just the ACSROOT variable. This saves a LOT of space.

As per ACS 10.0 this problem is fixed, the hardcoded path in omniORB was removed, the path required by omniORB is generated dynamically in runtime according the variables set in your environment.

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?](#)