



If you are using Vim as text editor [this cheat sheet](#) might be useful.

The first thing required is to prepare the environment for the compilation and execution of ACS and the project being developed.

#### Environment Preparation

```
source /alma/ACS-2019DEC/config/.acs/.bash_profile.acs
export INTROOT=~/introot
getTemplateForDirectory INTROOT $INTROOT
exit

#This could be added to the .bash_profile or .bashrc files
export INTROOT=~/introot
source /alma/ACS-2019DEC/config/.acs/.bash_profile.acs

#To create a directory for your module
getTemplateForDirectory MODROOT_WS <NameOfYourModule>
```

As part of what is delivered, there is a directory called ICD, where all the base definitions are given (ICD/idl) in IDL format (CORBA Interface Definition Language). These will generate code for the different languages supported by ACS (C++, Python and Java).

#### Environment Preparation

```
#Assuming the environment is correctly prepared
cd $REPO_PATH/ICD/src
make all install
```

This will generate the required libraries (.so and .jar) and source code (.h and .py) for the client stubs (definitions and communication) and skeleton infrastructure (server implementation base).

Another part of the project that is given, is the configuration of the system. In production this configuration is retrieved from an Oracle DB, however for these projects we're using a file system database based in XML files.

#### Environment Preparation

```
#Either execute the following in every console or add to your .bash_profile or .bashrc files
export ACS_CDB=$REPO_PATH/ITS/config
```

This will be used at run-time to deploy components in the running containers.