

2022-08-04 Spack Stack

Dom Heinzeller gave a presentation on the new SPACK-based methodology for building the JEDI software stack. This work is part of a joint effort with NOAA and potentially more partners to use SPACK (<https://github.com/spack/spack>, developed at LLNL) to build various software stacks. This joint effort is named "Spack Stack" and the software is currently held in a repository at NOAA EMC (<https://github.com/NOAA-EMC/spack-stack>).

Dom presented the following slides which provide motivation and details about the Spack Stack project.



After the slides were presented, the floor was opened up for Q and A.

Chris Harrop asked how the spack-stack methodology handles multiple versions of libraries where different applications have dependencies to these different library versions. Currently, the installation of multiple versions of the same library in one software stack is not fully supported. (For the most part this works, but there still exist some particular issues with module environments and container builds) We are working on solutions for these now. Another thing to note is that we want the spack-stack releases to operate on a cadence (monthly, quarterly, for example) and support for new library versions will be released on this cadence. We will need to optimize the best frequency for the releases to get a good balance between response time and test coverage.

Tom Robinson noted that it would be good to establish working meetings with partners which should help get solutions more quickly for the issues noted on the final slide. He also asked if we were settled on the name "spack-stack". Dom noted that the name "spack-stack" isn't cast in stone yet, and there are plans to create a new github organization to house this project in order to reflect that this is a joint effort between multiple entities. At the point of moving to a new github organization, the project could be renamed.

Ricardo Todling asked for clarification using different versions of ESMF (see the above question from Chris Harrop). This is handled properly given that you place the different versions of ESMF in different environments (so that any given environment contains only one of the versions of ESMF).

Matt Thompson asked how often do we plan to sync up with the LLNL spack repo. (For development purposes there exists a fork of the LLNL spack repo at NOAA.) The plan is to import new versions from LLNL at regular intervals, and to send back updates to LLNL (pull requests) as needed. Some of the codes are private so they cannot be sent back to the LLNL repo.

Chris Harrop asked about the process to change compiler versions, for example how difficult is it to do this? In theory, you should be able to make a copy of the spack-stack environment, change the compiler spec and hit the button to rebuild. In practice however, this tends to be easy when upgrading to a newer version of the same compiler, and not as easy when changing to a new compiler.

Matt Thompson noted that spack can get carried away and build more than necessary. For example it will build perl when it can't find the system installed perl. This is what the repos/site configuration in spack-stack is for. This configuration contains the specs for the pre-built system utilities on each HPC platform (which informs spack where to pick up the pre-built version, thus avoiding having to build it).

Chris Harrop suggested creating a slack channel for spack-stack. This could be useful, and it was noted that spack already has a slack channel.

Tom Auligne recognized the great team work that has brought about the spack-stack project. He also recognized all of the hard work from everyone toward getting the first release of Skylab out the door. Thanks Everyone!