

2018-09-06

The meeting opened with an update from the Boulder team (Xin, Mark) on issues that we have been having with the Singularity container. In the current version of the container, ufo-bundle builds but many tests fail with an error that looks like "No CommFactory for [serial]" or "No CommFactory for [parallel]", depending on whether the test is run with or without MPI. This has been traced to the definition of the Comm class in eckit but we do not yet have a fix. Xin and Mark are working on it and will keep the team apprised.

Chris H

Reported that ufo-bundle is working on Theia with intel compilers and develop-next version of oops. However, fv3-bundle is still having problems on Theia. It compiles but fails many tests. Chris is working on it.

Steve H

Is finishing up a ufo branch that includes the unified April 15 observational data sets for testing. ufo-bundle is now working with updated tests (including radiosonde, aircraft, and amsua) and he is now working on fv3-bundle tests. He expects to issue a pull request on ufo soon.

Anna

mentioned that there may be an issue with the April 15 GeoVaLs data for radiosonde. There may be inconsistencies on units for Log(P) and T. She is looking into this.

A question was raised about whether ODB can be used for 3Dvar and other DA runs. Steve H replied that the ODB functionality that is currently in JEDI is mainly a proof of concept with radiosonde data as an example. It is not ready for full DA applications but development is proceeding. There will be a meeting next week to discuss how to proceed with the ODB implementation.

Question posed to Dan from Mariusz about new amsua functionality. Dan responded that there were hacks put in to get it to work - the code needs to be cleaned up before it is ready to merge

Rahul

is working with Yannick on a model interface in oops that will allow users to start a model from a file or from an alternative initialization. He is also working with Guillaume on cleaning up the SOCA interface to include only MOM 6 and CICE 5 and to develop a workflow for SOCA.

Dan

is working on finalizing the B-matrix code development from the recent sprint. He is also working on solving the nonlinear balance equation with an improved Poisson solver. He is considering a finite element Poisson solver.

A question was raised about the ropp branch of ufo and Yannick responded that this was an experiment.

More generally, it was noted that there are many current branches of ufo and some of these are likely to be merged soon. Some of these branches are the product of the recent GNSSRO code sprint. There is also the fullob branch, which is awaiting the complete integration of April 15 test data.

UK Met Office

Marek reported that he is close to a pull request on the B-matrix branch of LFRic that arose from the recent code sprint.

He also asked about the size of CRTM. Ben responded that, although the size of the entire code+data files can be on the order of 74 GB, users should not be getting this when they clone the GitHub repository. The cloned repository should be only 2GB and the CRTM team is working to make it even smaller by replacing the binary data files with NetCDF.

Yannick mentioned that we are currently considering how best to handle large data files. One option is to store them on the cloud (e.g. Amazon S3) and then download them only as needed, when the relevant tests are run. The JEDI team would provide scripts or other tools to facilitate this.

Chris asked about version control in such a scenario.

Ben added that the plan for CRTM is to have one repo, crtm-dev, for the code and a separate one for the data files that uses git-lfs. This would address the versioning issue.

Marek then asked about the use of the term "control variables", which can mean different things to different people. He advocated for a common language. Yannick agreed that this can be a problem and welcomed suggestions.

Then HuL reported that there will be a meeting this afternoon to discuss how to proceed with merging ufo branches that arose from the GNSSRO code sprint. In addition to the application-specific features in ufo, they also added routines to calculate the geopotential in fv3-jedi and they made plans to discuss this further with Dan. Yannick encouraged team members to also develop similar functionality for other models, including MPAS and LFRic.

EMC

Guillaume mentioned that he had to comment out a line in the oops CMakeLists.txt file in order to get ufo-bundle to compile in the container:

```
ecbuild_find_mpi( COMPONENTS CXX Fortran REQUIRED )
```

There was some speculation as to whether this had anything to do with the container problems referred to above that Xin and Mark are working on. Subsequent investigation suggests that this is not the source of the problem - this line is already commented out in the develop-next branch of oops.

Guillaume is now working with Benjamin on integrating the bump interpolation into SOCA and getting it to work with Intel compilers.

Yangui expressed interest in meeting with Yannick and Andrew Collard to discuss Quality Control of data and, more specifically, where QC checks should be included in the JEDI repositories. Yannick responded that the QC infrastructure is not ready but needs to be developed. Dan and Rahul commented that GMAO is also interested in this issue and offered to help.

Steve S from the Met Office closed with a few more questions. First, he referred to a discussion we had yesterday during a models meeting about what grids variables should be on for Data Assimilation. Yannick summarized the outcome of the discussion by saying that all variables should be collocated on the same grid (both horizontal and vertical) for DA applications, at least for now. This is the simplest situation - we need to start with this and get it to work. In the future we can consider extending this to staggered grids but this is a substantial effort.

Steve S then asked about the installation of software - what goes into the build of a bundle as opposed to being installed separated as part of the container or otherwise part of the environment. For example, eckit and fckit are now part of the Singularity container along with other external software such as boost. But, packages like crtm are built as part of a bundle. One could in principle include crtm in the container/environment. Yannick responded that there is no set rule for what goes into the container and what does not. External applications like boost are beyond our control so it makes sense to include them in the environment whereas crtm is being developed along with JEDI so including it in the bundle helps you to keep up with the latest updates.

To some extent, this is up to the user. Experienced users can install libraries and set up their environment as they wish whereas those users who do not wish to delve into the technical details should be able to get up and running with minimal difficulty by using the container and bundles.

This led to a more general discussion about what the Singularity container is used for and whether users need to install it themselves. The answer to the second question is yes - users are responsible for installing singularity on their systems, if it is not already there and if they have root permission: for further details see: [the JEDI documentation](#). After Singularity is installed, the user can simply download the JCSDA container as an image file and then enter it (i.e. execute the application). Regarding the first question, Mark and Yannick outlined three main functions for the Singularity container:

1. As an **implementation tool** to help users get up and running with JEDI as quickly and easily as possible. This is intended in particular for new users and for scientific users who are interested in exploiting the functionality of JEDI without having to deal with the technical details of the software implementation.
2. As a **touchstone** to facilitate code testing and debugging. Having a common environment (including identical versions of compilers, libraries, etc) can help developers rule out a number of potential issues when trying to reproduce and diagnose a problem.
3. As an **access point** to particular models. For example, a developer may develop a new feature for FV3 and then may wish to extend to other models such as Ifric or mpas. A container can help them get those other models up and running with minimal effort so they can test their new feature.

For these reasons, we expect to continue to support and develop the JCSDA singularity container in the future. However, as we gain experience with cloud computing platforms such as Amazon, some of the functionality listed above may shift at least in part to the cloud.