

**Present:** Amy, Andrew, Claire, Julien, Mike, Pierre

**Apologies:** Jerome, Seb, Simon

### 1. The **definition of the scope** of V&V

This has been clarified somewhat but the objectives need to be clarified further. The IMMERSE test cases seem to sit in a “grey zone” between verification of correct implementation and validation of improved performance. This issue has come up in the work Jerome and Laurent are doing within IMMERSE to set up a simple verification demo test case (the demo generates a small file as output that is tested to generate a red/green flag).

Julien suggests that we need to build the “vision” further of the structure and framework for the types of V&V. He suggests the vision should be aim to “shorten” the release process for small changes to the code or the tests conducted on it. He suggests four levels of testing: 1) unit testing; 2) SETTE testing; 3) simple demo configurations and 4) validation of real-world predictions against observations (conducted by Mercator Ocean International, CMCC and Met Office)

**Action:** Mike, Julien and Amy to further clarify/articulate objectives and scope of V&V

### 2. **Tools: SETTE**

SETTE is a simple home-made tool used by developers. For a given configuration SETTE determines whether it will compile & run and whether the results in certain files are the same as in an earlier version. The general view is that it is easy to use and maintain and (generally) does what we need. It consists of two parts. A part to compile and run NEMO and a part “validating” the outputs.

A number of Pierre’s ideas for improving SETTE were discussed:

- Make the submission script more generic; this will make it more maintainable
- Enable parallel compilations (this task would need some scoping)
- Make the validation part easier to use (?) and extend it to include more simple demo cases
- Enable SETTE to walk through permissible permutations of options

The view was that SETTE should not be developed too far: it should remain a simple tool to use and maintain.

**Action:** Pierre to add his proposals to the V&V web page and pass the SETTE section to Andrew for his input.

### 3. **Tools for continuous integration**

**Trusting** has been developed by Nicolas Martin. It is based on a tool developed by IPSL. It was run for 2 years then turned off. It provides summaries of results in html format. It is not based on the SETTE tool but the two tools could be rationalised. Claire thinks that Trusting is portable and its dependencies could be captured in the Arch files. The “real work” is to define what verification/validation tests should be performed.

Trusting will be set up to run continuously on release 4.0.head before the end of this year. Trusting and SETTE now have more similar capabilities and it may be that fewer issues will be revealed (i.e. problems to solve generated) by Trusting than when it was last used.

Trusting should be considered to be a demonstrator of continuous integration. There are a number of commercial tools such as Jenkins designed to support continuous integration. Some of these tools provide measures of the coverage of the tests of the code. Installation of these tools on our systems can be problematic. But the tools are available through web services and might be used with containerised versions of NEMO (Italo is exploring containers for NEMO). Payment for web services would have to be organised.

**Action:** Claire to update the section on Trusting and continuous integration (noting recent experiences with video-conference merges) and discuss it with Julien and Nicolas (when he returns).

#### **4. svn/trac and git/github**

Transitioning to git would have a number of costs (e.g. re-training, temporary loss of productivity). Benefits include the facilities for team communication within github and access to services for code testing. We need to consider this issue on a 10-year time-scale. Some systems (including FESOM?) use both svn and git. The Met Office is considering transferring from svn to git for its UM/LFric system.

**Action:** Claire and Amy (?) to summarise pros/cons of transition to git and organise a discussion with the NST.

#### **5. Unit testing**

Unit testing is usually done when a code is first developed. The cost/benefit of introducing it retrospectively is not clear. MPAS (Mark Peterson) and FESOM (?) have systems for performing unit tests and employing them in the regression testing at a module level. If a framework for such unit tests were developed it would need to be implemented module by module.

**Action:** Mike and Simon to scope out options for introducing unit testing.

#### **6. Time-scales**

Some of the work on SETTE could usefully be included in next year's NST work-plan. The initial scoping for that would need to be done by Sept/Oct this year. The next NEMO Development Strategy is likely to be developed next year. We want to have a reasonable version of this document finished before that discussion starts. Only the git/github issue might benefit from delay. It was agreed that we should meet again to review progress in September.

**Action:** Amy to doodle poll a date to meet.