Ocean Sci. Discuss., 12, C1204–C1207, 2015
www.ocean-sci-discuss.net/12/C1204/2015/

Ocean Science
Discussions

Interactive comment on "Technical note: Harmonizing met-ocean model data via standard web services within small research groups" *by* R. P. Signell and E. Camossi

Anonymous Referee #1

Received and published: 17 November 2015

This manuscript describes how data services can be implemented at a local level to help with data management and distribution within individual research groups. The paper promotes practices that are rapidly becoming "standard" but thus far are typically deployed at larger data/modeling centers. While I applaud this goal, I think the manuscript needs some work before it can be published.

I think at some level the paper lacks focus. It's not clear to me what the main theme is, e.g., whether it is advocating smaller modeling groups to setup THREDDS servers, or it is attempting to show how straightforward it is, or is it trying to show what the benefits (by way of example) are in doing so? I suspect the authors are trying for all three, but

the result is none are fully explored. It certainly would not be possible to set up such a system solely based on this manuscript. Likewise, it's not quite clear what such a system would look like or concrete examples of how it works.

Further, there appears to be a mix between fine detail, e.g., the use of datasetScan, and more big-picture types of descriptions that are not fully explained, e.g., "use pycsw". I would recommend the authors try to key on one issue and fully develop that. It should be noted, however, that such a paper could become more like a "how to", or user guide. Instead, I think showing concrete examples, perhaps case studies with and without data services, would help.

Some more specific points: 1. The paper uses a lot of acronyms, out of necessity, but only occasionally (and almost randomly) are these spelled out. In some cases it might be obvious, e.g., NATO. On the other hand, it's probably good practice to either spell them all out on first use or include a list of acronyms at the end. For example the asbtract alone has netCDF, THREDDS, pycsw, NATO and USGS; OGC on the other hand is spelled out four different times.

2. It might be a minor, or sensitive point, but I prefer the phrase "model output" over "model data". If nothing else, this would help with potentially complex phrases that include "model data" and "data model", such as lines 24-25 "model data infrastructure ... for data models". This is just a suggestion, not a criticism.

3. The authors don't give any measurable quantification of using a data service instead of a regular file system. Is there a way to produce usage metrics, for example, with THREDDS that can't otherwise be done (thus giving the data providers a better idea of who is using the output)? Or metrics on access speeds? In other words, if I have a small modeling group and don't really care about data discovery issues, why would I go through the trouble? it is faster? Can I more easily track users?

4. THREDDS has a limitation that input data must be in netCDF format. While netCDF is certainly a standard, what happens if modeling groups produce output in mutliple

formats, e.g., grib and/or flat binary? Would they have to setup another OPENDAP server?

5. As far as I know, THREDDS will require an apache tomcat server. It's not clear what sort of requirements this puts on the server machine. For example, groups may not want to overload a production machine (running models) with a data service that could overwhelm the machine resources. Or, perhaps this is not a drain on the machine memory and/or CPU?

6. The abstract describes data services in a somewhat independent way, e.g., pycsw is used for data discovery, THREDDS for data delivery, etc. However, the situation is more parallel (I think). It all starts by having netCDF data. TDS and NCML then expose these data via OPENDAP to client tools. At the same time, TDS creates ISO metadata records that can be harvested by pycsw. And, TDS can be configured with a builtin tool providing data browsing capabilities (WMS).

7. Any comment on the advantage of pycsw over the other CSW metioned on page 4 (lines 15-18)?

8. Section 3.1, machine resources needed for TDS? Want to add ability of NCML to "modify" output, e.g., hide variables, rename, add metadata, etc.?

9. Section 3.2 might be cut. It's too short to be meaningful; maybe add discussion into 3.1, e.g., "opendap enabled tools".

10. Ditto section 3.4 (WMS not fully explained).

11. Include figure for section 3 that shows integration of these? Maybe of Godiva2?

12. I'm not sure I understand the bottom paragraph on page 7; "During the trial"? Using GeoServer not TDS? CKAN not pycsw?

13. The example display of glider output is interesting, but not really in line with the main theme of model output and data services. In addition, it opens a lot of questions,

for example, how the lat/lon/depth are interpolated and deconvolved with time. I think with such color-shaded plots it is assumed that the glider up/down is done instantly in time? Otherwise is it better to display these as saw-tooth tracks (up/down)?

14. Section 4 introduces Ipython notebook, which is very interesting, but somewhat outside the scope of the rest of the paper.

15. Section 4.2 is also somewhat cursory; it gives very specific details about the TDS implementation at USGS. Could this be re-written to include a more specific account of a) what was needed; b) how it was implemented; and c) what the benefits are?

16. In Discussion, mention other benefits, such as proper cataloging of model runs, exposure to other TDS catalogs, "standarization" of output, etc.

Interactive comment on Ocean Sci. Discuss., 12, 2655, 2015.