

NCLADT generated TIM talking points/questions

11/21/08

1. GFE's AFPS

- a. Will not exist in awips2.
- b. There are 48 unique AFPS calls in heavy use in the field
 - i. By 4796 apps -- 2044 unique apps at 121 sites
- c. What are their awips2 equivalents?

2. Status of SmartScript?

- a. 151 methods in the awips1 version of SmartScript.py
- b. 110 methods in TO9's SmartScript.py and Procedure.py
- c. Many are not fully implemented

3. ObjAnal.py

- a. How to make STR utilities play?

4. NetCDF replacement approaches

- a. py scripts
- b. pl scripts

5. Procedures

- a. How do they work?
- b. runProcedure.sh - relies upon AFPS

6. On the final, deployed awips2 system

- a. Will the awips1 runtime environment exist?
 - i. Will /awips/fxa exist?
 - ii. Will /data/fxa exist?
 - iii. Will /awips/hydroapps exist?
- b. MHS will not be re-implemented for awips2? If so, this means that handleOUP.pl would work as coded if the awips1 runtime environment were left as it is. "awips1 runtime environment" =
 - i. The rc scripts that set environment variables,
 - ii. The /awips and /data directory structures
 - The programs in /awips/fxa/bin
 - iii. handleOUP.pl calls distributeProduct. Will awips1's distributeProduct exist on the awips2 system?

7. Will we be able to use the CLI to retrieve grids?

8. Hydro apps

- a. Will the hydro databases exist as they do in awips1 (same name and schema)?
- b. Fed by the same ingest stream, will they contain the same data as in awips1?
- c. Apps that do psql reads from any awips1 hydro db table should be able to do the same in TO10 without modification. (verify)
- d. Will psql writes be allowed?
- e. How will CHPS and awips2 interoperate?

9. Slide 3 of the TO10 TIM doc says that the CLI tool "shall include a command line interface to the uEngine script runner". So I guess we'll be able to pull grids from edex with it. How will the output will be formatted?

10. Elaborate on Raytheon responses to Govt. GFE questions

- a. **Govt. Comment:** Each method of awips1's SmartScript.py gets used by some STR app at some site -- there is no SmartScript.py method that is not used. I see that some of awips1's SmartScript methods have been moved to Procedure.py in awips2

RTS Response: The procedure methods have been moved back into SmartScript in TO10. Long story short, I was hoping to move from the GFE python style of everything inheriting everything, and move to a more modular design, but upon learning that there's so many custom procedures out there, that won't work when we're aiming for backwards compatibility. So those methods are back in SmartScript.

- b. **Govt. Comment:** Utilities like the STR's ObjAnal, if dropped in the utilities directory, don't seem to get picked up by the system.

RTS Response: I wasn't aware of that. The refactor of how smart tools work (single python interpreter) in TO10 should take care of this.

- c. **Govt. Comment:** AFPS is not used only by SmartScript, but is called directly by many Tools and Procedures in the field.

RTS Response: AFPS is incompatible with AWIPS II, no way around it. That said, for most of the methods used by AFPS, there is either an equivalent Java or Python method that can be used to accomplish the same thing. It's fairly straightforward, but if you find you need help translating a former AFPS call to the right call in AWIPS II, let me know.

- d. **Govt. Comment:** preProcessTool and postProcessTool need to be able to accept a variable length arg list

RTS Response: In TO10, the pre and post process methods work very similar to how they worked in Tool.py. That is, they retrieve the method's argument names, and then use getArgValues() to get the argument's values.

- e. **Govt. Comment:** How do procedures work? How is runProcedure.py run?

RTS Response: A new Java class in TO10 named GFEProcedure runs procedures. Running a procedure from the command line or a cron job (i.e. not CAVE), will not work at this time.

- f. **Govt. Comment:** callSmartTool and callProcedure don't exist in awips2's SmartScript.py

RTS Response: I've remedied that in TO10, successfully ported and tested.

- g. **Govt. Comment:** What is "grid access"

- h. **Govt. Comment:** The ability to create, query, and archive gridded datasets outside of the default areas is vital. This piggybacks with the above concerns. It's ok if we don't use netcdf, but we need to have something to store to. For instance, ObjAnal reads in a series of observations points and interprets them to a uniform grid. This grid is then saved. This grid is used both to check forecast verification in the future, and improve the short term forecast in the operational GFE.