

SAM packaging status report: SAM client freezing

Alan Sill

SAM project planning meeting

Apr. 7, 2005

Work list

- No repackaging - use fnkits ✓
- Work for both CDF and Dzero ✓
- Install as sam into sam's ups databases ✓
- Products version control (manual only now) (*)
- Upgrade/downgrade ability ✓ for client
- Inquire for differences between versions (?)

Work list cont'd

- Set up specific sam version (have already) ✓
- Preserve and restore local configurations --
NEEDS DESIGN - BIG PROJECT!
- Separate client/server installations; different modes for private vs. public networks (?)
- Compact client capability ✓ (This report)

Auxiliary work list

- These issues emerged as sub-issues of the above, and as we began/continued to work:
 - Integration tests for release candidate version sets: **NEED MGMT DEFINITION**
 - Role of testing dbserver??
 - Method of selecting dbserver??
 - vdt upgrade ✓ Done

This talk: client freezing

- Nov. 2004: first idea; Dec: proof of concept.
- Jan. 05: Expert could freeze client with much tedious manual assembly, but fragile against included product changes. Results highly experiment-specific w/ manual versioning.
- Feb., Mar.: Separated configuration from frozen client. Automated script to produce the client. Made result stand-alone w/ no external env-var. setup required. Tested extensively for Dzero + CDF farms. Added new “sampy” functionality.

Status as of April 7

- Have extensively rewritten the internals of the sam_freeze product components and made them into a single ups-able product.
- Will be able to simply *upd install sam_freeze; setup sam_freeze; sam_freeze my-destination-dir.*
- Goal: Put control of production of the sam frozen client back into hands of developers.
- Side effect: sam freezing can be used as a debugging tool, for specialty customization, etc.

The freezing process

- `cx_freeze`, `cx_pygenlib` have been combined into a new single “`sam_freeze`” product with base executables and a `sam_freeze` script that is customized to sam.
- Freezes whatever sam you have working.
- Self-discovers `sam_user_pyapi` commands and produces needed import placeholders.
- Automatically includes proper omniORB libs.
- Copies result to user-specified directory.
- Creates softlink for `sampy`.

sampy: new method for access to SAM client api

- Goal: provide access to python api; needed for JIM, also useful for SAM scripting.
- Access was thought possible via “zip file” mode for python imports, but only in recent versions of Python (≥ 2.3 , 2.4 preferred)...
- Workaround (idea due to Robert Illingworth): use argument parsing to determine whether frozen module is being called to run a script.

Example of sampy feature

```
sammy my_client_script (args)
```

- Python and any modules available in the frozen sam client are available to the script
- Allows direct use of sam python api
- Argument stack unaltered (but note: `arg(0)` is now “sammy” so be careful of ordering)
- Script does not need to be named “.py”
- Runs totally stand-alone, does not depend on external python, libs or products.

Testing

- Testing for Dzero done by Robert -- much valuable feedback.
- Eliminated experiment dependencies.
- Tested for CDF farms by Suen Hou, Tsan Hsieh and Fedor Ratnikov. Minor adjustments to SamStoreCdfFile and `_v6` done by Fedor to work with sampy.
- Passed all major tests so far.

Summary of new features

- SAM frozen client is totally stand-alone, distributable in its own frozen tarball (or by ups), depends on no external products and has no special environmental setups.
- Access to sam python api is possible for external scripts through the sampy command. This is also stand-alone and uses sam itself.
- Freezing has become “push-button” and can be put back under developer control.
- `sam_freeze` ups-able product created.

Next steps:

- Address assembly of the sam_products, sam_client_products, and sam_dbs_products table lists.
- Finish ups product rollout.
- MGMT: Determine process for integration testing and selection of product sets in released versions.
- Goal: Find a product packaging method that will make it easy to deploy SAM products on the grid. *Personal preference: use Pacman.*