CODESH

COllaborative DEvelopment SHell

An Intelligent Development Shell for Seamlessly Logging, Exchanging and Reproducing Results and the Methods Used in Obtaining Them

Dimitri Bourilkov, Vaibhav Khandelwal

University of Florida

WMSCI 2005 July 10-13, 2005 - Orlando, Florida, USA

Enormity of Data and Virtual Data

- Most scientific data are not simple "measurements" ⇒ produced from increasingly complex computations (e.g. reconstructions, calibrations, selections, simulations, fits etc.)
- Large size – Petabytes and Exabytes.
- HEP (and other sciences) increasingly CPU/Data intensive: **Programs and how-to** become a vital intellectual resource of the scientific community ⇒ ***need new ways to collaborate***
- **Virtual data** are data products with a ***well defined method of (re)production***; "virtuality" with respect to existence ⇒ can define data products for future production or record the "history" of products that exist now or have existed in the past

  Log data provenance by tracking how new data is derived from transformations on other data

We already do this, but manually!

Virtual Data Motivations

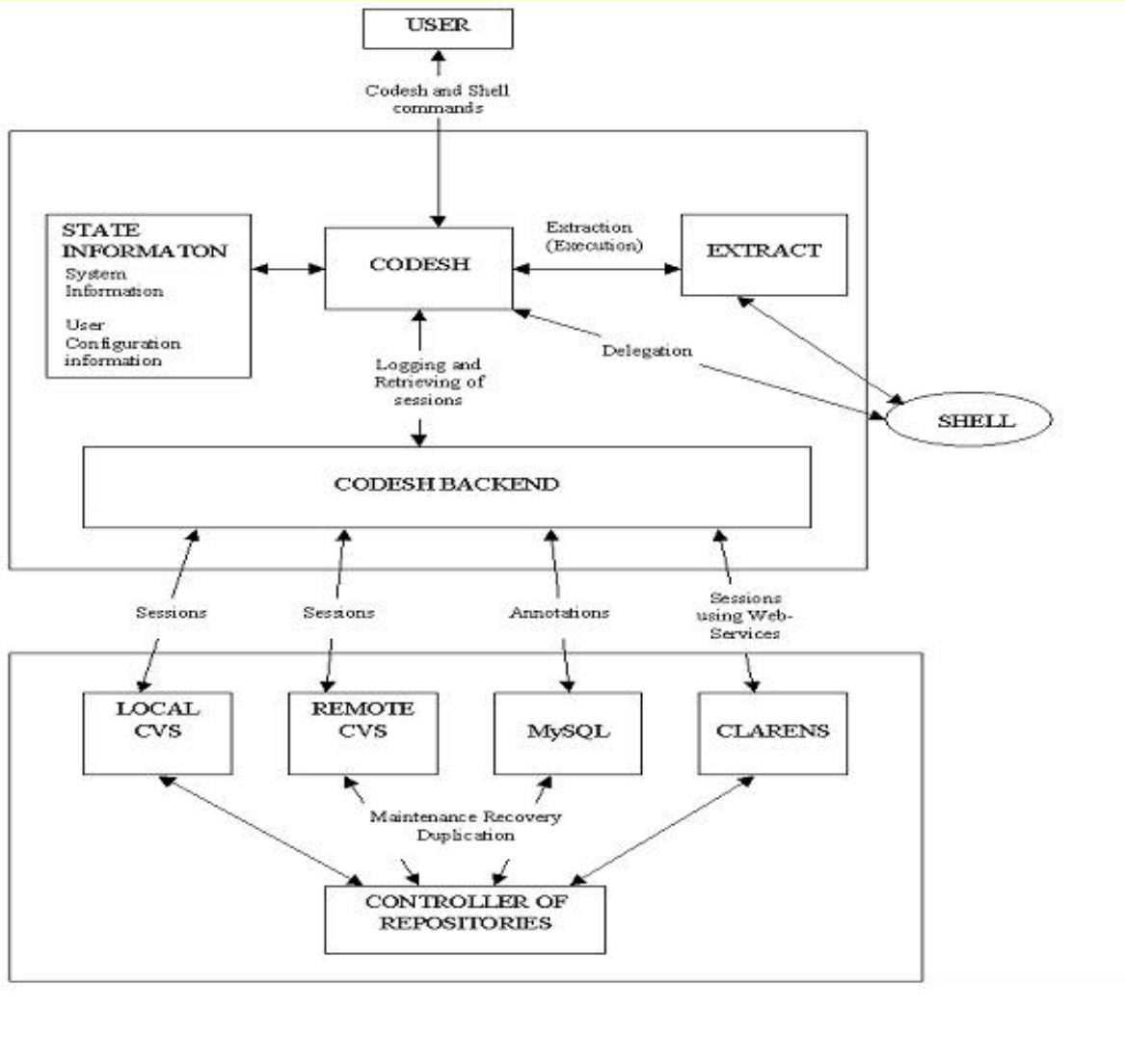
- Data **track**-ability and result **audit**-ability: "Virtual Logbook"
  - In the nature of science
  - Reproducibility of results
- **Tools** and **data** sharing and collaboration (data with "recipe")
  - Individuals **discover** other scientists' work and build from it
  - Different Teams can work in a **modular, semi-autonomous** fashion: reuse previous data/code/results or entire analysis chains
- **Repair** and **correction** of data – c.f. "make"
- **Workflow** management, **Performance** optimization**:** data staged-in from remote site OR re-created locally on demand?
- **Transparency** with respect to location and existence

CODESH

- Concentrate on the interactions between scientists collaborating over extended periods of time
- Seamlessly log, exchange and reproduce results and the corresponding methods, algorithms and programs

  Automatic and complete logging and reuse of work or analysis sessions (between checkpoints)
- Extend the power of users working or performing analyses in their habitual way, giving them virtual data capabilities
- Build functioning collaboration suites (stay close to users!)
- First prototype uses popular tools: Python and CVS;

CODESH Architecture - Scalable and Distributed



CODESH Architectures – CVS Server Backend

- **Sandbox programming** – work on per session basis
- CVS provides **version control** by tagging releases
- CVS **tags** act as **unique IDs** for virtual data products (the namespace can be structured by a collaborating group e.g. one big cave or many barrels in a cave, selected on a session basis)
- Both **local** and **remote** modes of working
- CVS pservers (secure, efficient remote stores):
  - Only CVS user accounts with password authentication, no UNIX accounts on the server (gridmapfile uses same idea)
  - read/write access control lists (per user & directory)

CODESH Architectures – Clarens Backend

- Open source, secure, high-performance, web-services based "portal" for ubiquitous access to data and computational resources provided by computing grids.
- **UltraLight** project - *An Ultrascale Information System for Data Intensive Research*
- **SOA** –Service Oriented Architecture.

Possible scenarios

Case1: Simple

User 1 : Does some analysis and produces a result with tag ***analX_user1.***

User 2: Browses all current tags in the repository and fetches the session stored with tag ***analX_user1.***

Case2: Complex

User 1 : Does some analysis and produces a result with tag ***analX_user1.***

User 2: Browses all current tags in the repository and fetches the session stored with tag ***analX_user1***.

User 2: Does a modification in the program obtained from the session of user1 and stores the same along with a new result with tag ***analX_user2_mod_code***.

User 1: Browses the repository, finds that his program was modified and decides to extract that session using the tag ***analX_user2_mod_code***.

This scenario can be extended to include an arbitrary number of steps and users in a working group or groups in a collaboration.

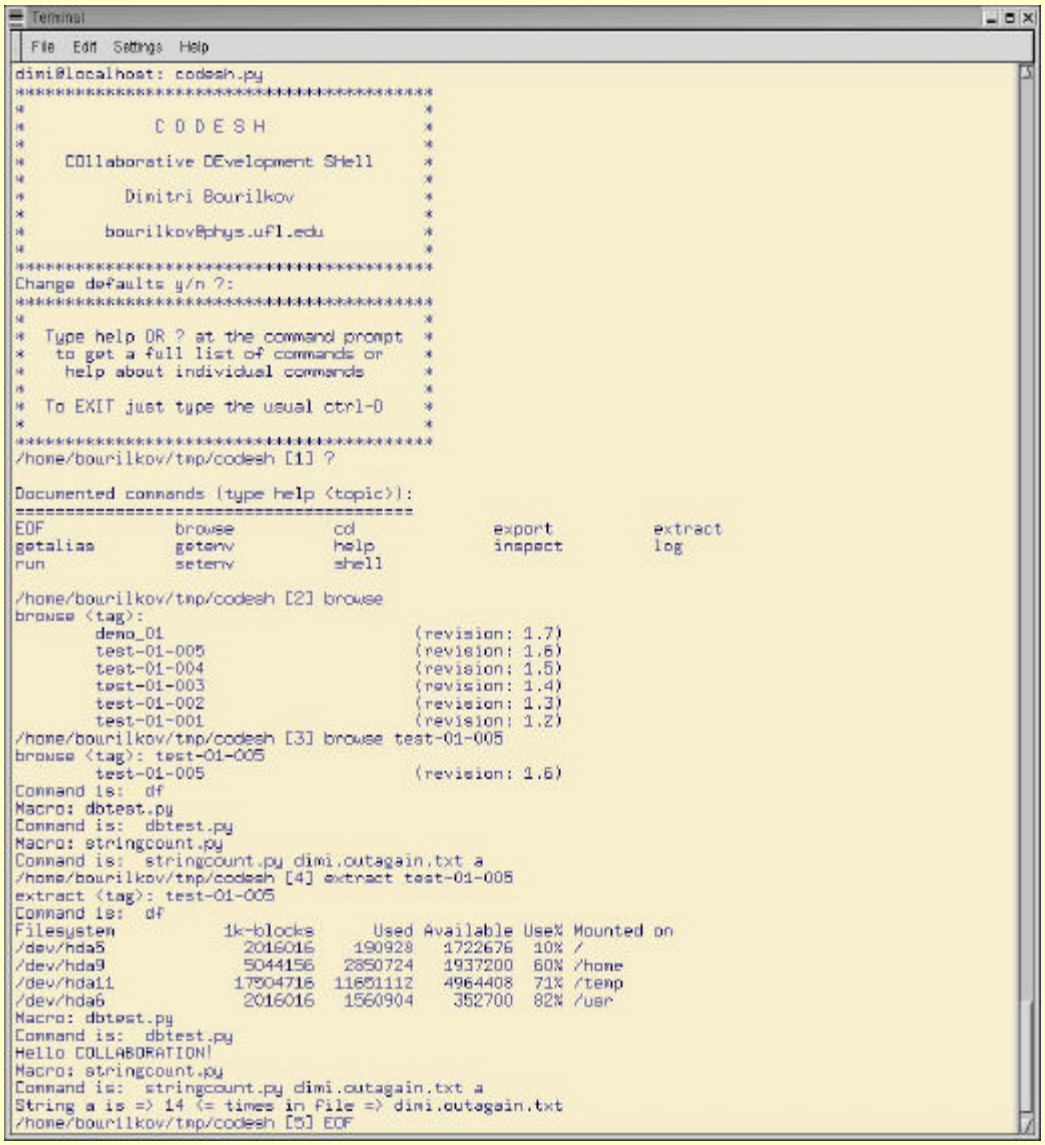
Annotations

- Log **annotations** (& possibly **summary results** or **metadata**) in the repository /data equivalence!/
- **Brief** annotations – use cvs –m “Annotation” ...
- Optional **complete** annotations - possibly MySQL servers for metadata (fast search for large data volumes e.g. with indexing)
- The users can browse (a subset of) the existing tags, inspect the annotations, and, if interested, reproduce the results (or just get the howto), modify them etc.

Extensible Command Set

- Session commands
  - open <session>
  - close <session>
- During analysis
  - help <command>
  - browse <tag>
  - inspect <tag> <b|c>
  - startlog
  - log <tag> <annot>
  - annotate <tag>
  - extract <tag>
  - snapshot
  - setenv, getenv
  - getalias
- Administrative tasks
  - copy <tag> <from> <to>
  - move <tag> <from> <to>
  - delete <tag> <from>
  - archive <tag> <to>
  - retrieve <tag> <from>

Working Releases



- Virtual log-book for “shell” sessions
- Parts can be local (private) or shared
- Tracks environment variables, aliases etc during a session
- Reproduce complete working sessions

Working Releases

- The client codes are easy to download and install (stored in CVS) – checkout and build
- All you need to run the clients is CVS and Python 2.1 or higher
- Users can browse the virtual data catalogs and reproduce the examples stored there
- They can play with e.g. new analyses and store them locally or on our server, or install new servers for groups collaborating on a project

Outlook

- Work in progress – **first** CODESH release out
- We are looking forward to **user feedback**
- Possible **future directions**:
  - Develop the Clarens backend completely
  - Extend remote data access: e.g. Clarens, xrootd …
  - Automatically convert session log to workflow
  - Tune on smaller samples, schedule on grid for larger tasks

A picture is better than 1000 words: Try out the releases !

CAVES white paper: arXiv: physics/0401007; http://arxiv.org/abs/physics/0401007.

CODESH / CAVES papers: 1) arXiv: physics/0410226; http://arxiv.org/abs/physics/0410226.

2) ICCS 2005, LNCS 3516, pp. 342-345, Springer Verlag, 2005.

3) WMSCI 2005, vol. VIII, pp. 175-179, 2005.

More info and all papers at http://cern.ch/bourilkov/caves.html

CODESH in Action

DEMO