Integrated Scientific Workflow Management for the Emulab Network Testbed

Eric Eide, Leigh Stoller, Tim Stack, Juliana Freire, and Jay Lepreau
University of Utah, School of Computing

...in Short Papers Session III, Sat., June 3 @ 2:00 PM

Problem and Approach

In the networking and distributed systems communities, the difficulty of managing experimentation processes and results is a barrier to high-quality research and education.

Today's networked and distributed systems are complex:

- 10s–1000s of devices
- many device types
- many programs
- allocation
- configuration
- coordination

Current testbeds are good at *managing the laboratory* but provide little for *managing the experimentation process:*

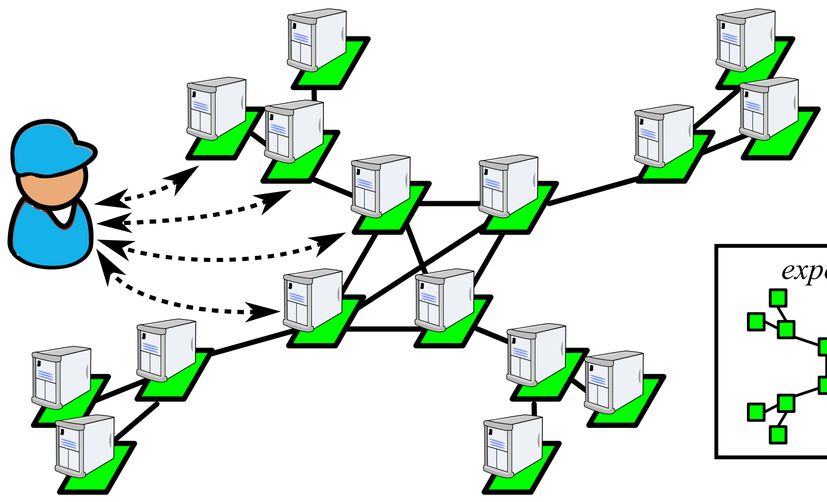
- instrumentation
- automation
- data collection
- analysis
- exploration
- collaboration
- reproduction
- reuse

We are expanding Emulab with concepts from scientific workflow management systems to create an integrated experimentation workbench for networked systems.

The Emulab Network Testbed

Emulab is our popular, free-for-use, Web-accessible, time- & space-shared, reconfigurable network testbed.

The current interface is focused on *testbed resources*, not *user workflow*.



Demand and opportunity: Many of Emulab's users have outgrown this model. They need better ways to organize, record, and analyze their work.

A New Model of Experimentation

Template: a parameterized description of resources & activities

Swapin: a container of testbed resources

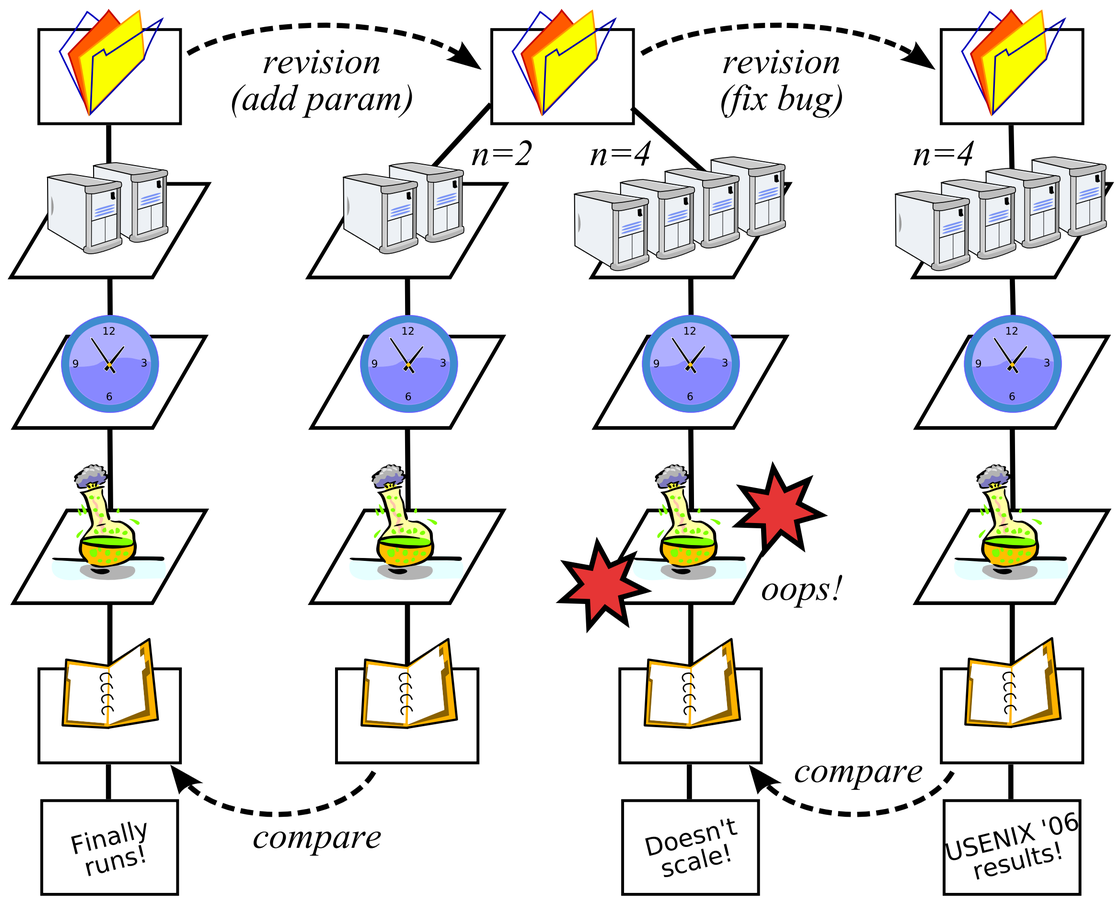
Experiment: a user-defined time period & container of Activities

Activity: a parameterized group of processes, workflows, etc.

Record: the "flight recorder" of an Experiment

Metadata: persistent annotations

Persistent and **Transient**



The challenges of...

workflow within...

Encapsulation: complete & precise, automatic & manual

Orchestration: support both directed & exploratory work

Data Management: collect data via probes, persist and analyze via a datapository

and workflow across.

Definition and Execution: separate notions of "description" and "run"

Grouping: navigation through "experiment space" & "result space"

History: let users "fork" and back up easily, without losing work

Implementation

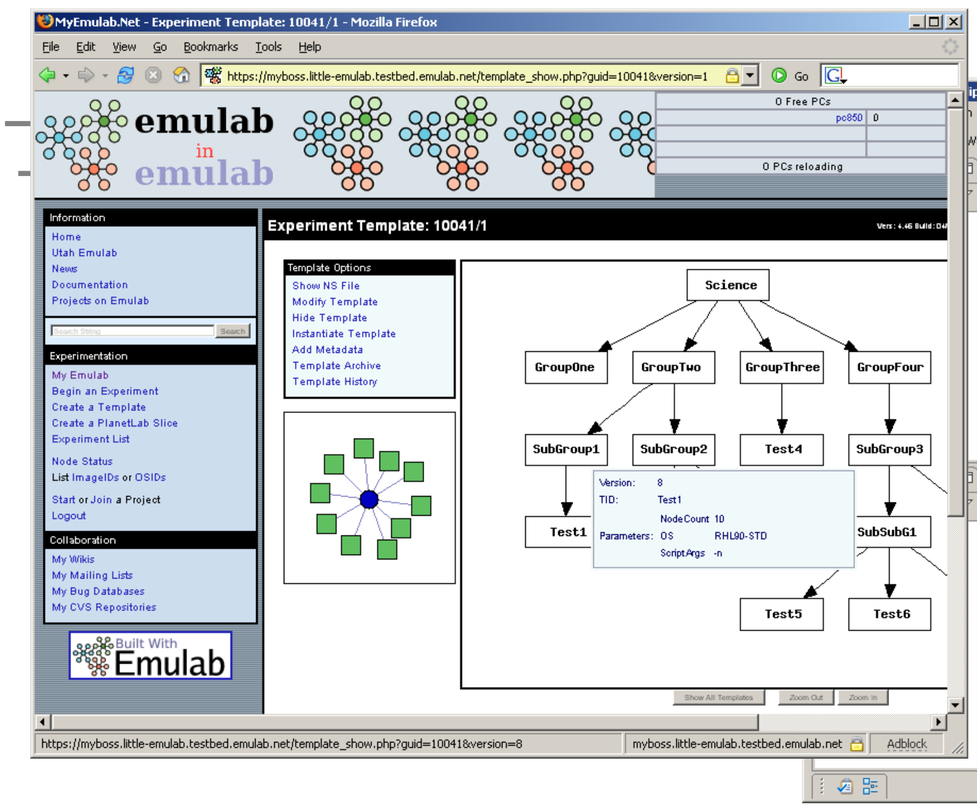
Web & GUI tools provide access to Templates, Swapins, Records, etc.

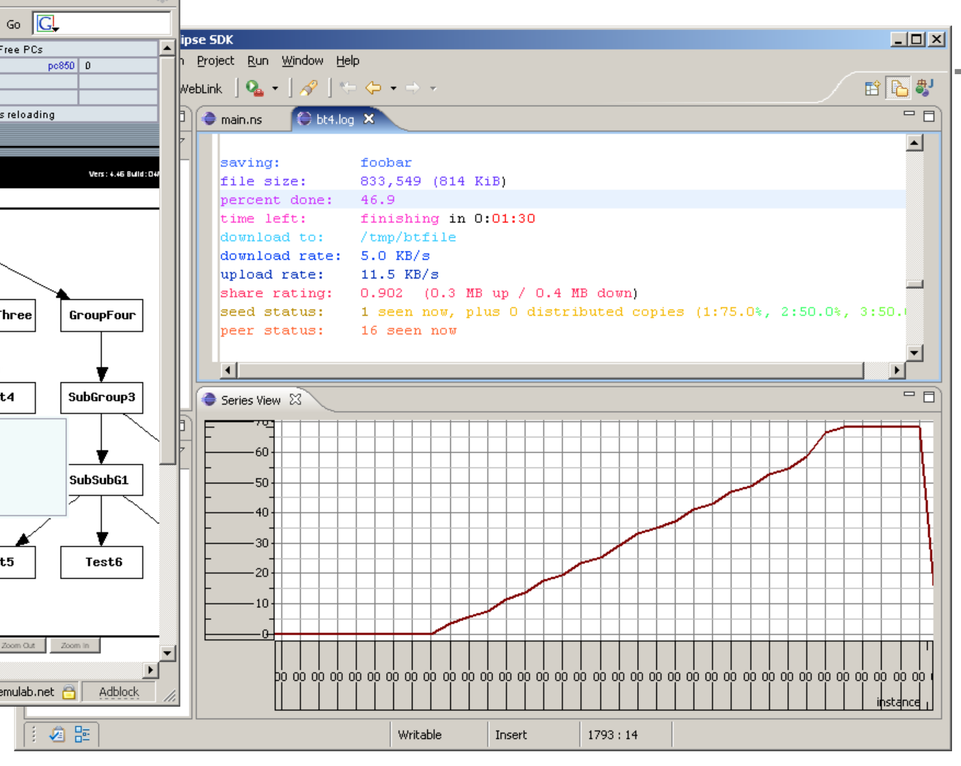
Network probes are transparently inserted to capture packets.

An NFS monitor analyzes traffic and identifies files for the Record.

"LogHole" manages and collects files.

A Subversion repository stores files for Templates and Records; **a datapository** stores database tables and records.





This material is based upon work supported by NSF under grants CNS–0524096, CNS–0335296, and CNS–0205702. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.