Liquid Benchmarks

Sherif Sakr[1] and Fabio Casati[2]

[1]NICTA and University of New South Wales, Sydney, Australia
and
[2]University of Trento, Trento, Italy

2^{nd} **Second TPC Technology Conference on Performance Evaluation and Benchmarking (TPCTC'10)**

17 September 2010

Problem Overview

- The last two decades have seen *significant growth* in the number of scientific research publications.
- An important characteristic of Computer Science research is that it produces artifacts other than publications, in particular **software implementations (prototypes)**.
- There is a continuous existence of *performance improvement claims* from researchers.
- The *quality* of reported experimental results are usually limited due to several reasons such as: insufficient effort or time, unavailability of suitable test cases or any other resource constraints.
- Researchers are usually focusing on reporting the experimental results of the *good sides* for their work which may not reflect the whole picture of the real-world scenarios.

Problem Overview

Liquid Benchmarks: Benchmarking-as-a-Service

- An open call for online platforms that facilitates applying **independent** *experimental evaluation and comparison* techniques between competing alternatives of algorithms, approaches or complete systems in order to assess the practical **impact** and benefit of research results.

- The main aim of **LiquidPub**[1] Project is to develop concepts, models, metrics, and tools for an efficient, effective and sustainable way of creating, disseminating, evaluating, and consuming scientific knowledge.

[1] http://liquidpub.org/

Benchmarking Challenges in Computer Science

Not enough standard benchmarks are available or widely-used

- A *benchmark* is a standard test or set of tests that used to evaluate/compare alternative approaches that have a common aim to solve a specific problem.
- A benchmark usually consists of a *motivating scenario*, *task samples* and a set of *performance measures*.
- Unavailability of a standard benchmark in a certain domain makes the job of researchers hard to evaluate/comprare their work and leads to having several *adhoc* experimental results in the literature.
- For any benchmark to be successful, it must gain *wide acceptance* by its target community.

Benchmarking Challenges in Computer Science

Not enough standard benchmarks are available or widely-used

- Designing a successful benchmark is a quite *challenging* task which is usually not easy to be achieved by a single author or research group.
- In practice, very few benchmarks were able to achieve big success in their communities (e.g TPC, oo7, XMark).
- In ideal world, simplifying and improving the task of building standard successful benchmarks can be achieved through *collaborative* efforts between peer researchers in the same fields.

Benchmarking Challenges in Computer Science

Limited repeatability of published results

- In an ideal world, researchers should make the source codes/binaries of the implementation of their contribution in addition to the experimental datasets *available* for other researchers to be reused for repeating the published results in their paper. **Debates**!

- Unfortunately, the world is not always ideal ;-)
  - XMLCompBench.
  - SIGMOD Repeatability Experiment.
  - VLDB Experiments and Analysis Track.

Constraints of Computing Resources

- In some domains, conducting experimental evaluations may require *huge* computing resources.
- Conducting experimental evaluations may require using *different settings* for the computing environments in a manner that is similar to different types of real-world environments.
- Such computing resources requirements may be not available for researchers in their home environments/labs.
- Achieving a *fair* and *apples-to-apples* comparison between any two alternative scientific contributions requires performing their experiments using exactly *the same* computing environments.
- In an ideal word, researchers should have access to *shared computing environments* where they can evaluate/compare their contributions. The suitable configuration of these testing computing environments can be also decided *collaboratively*.

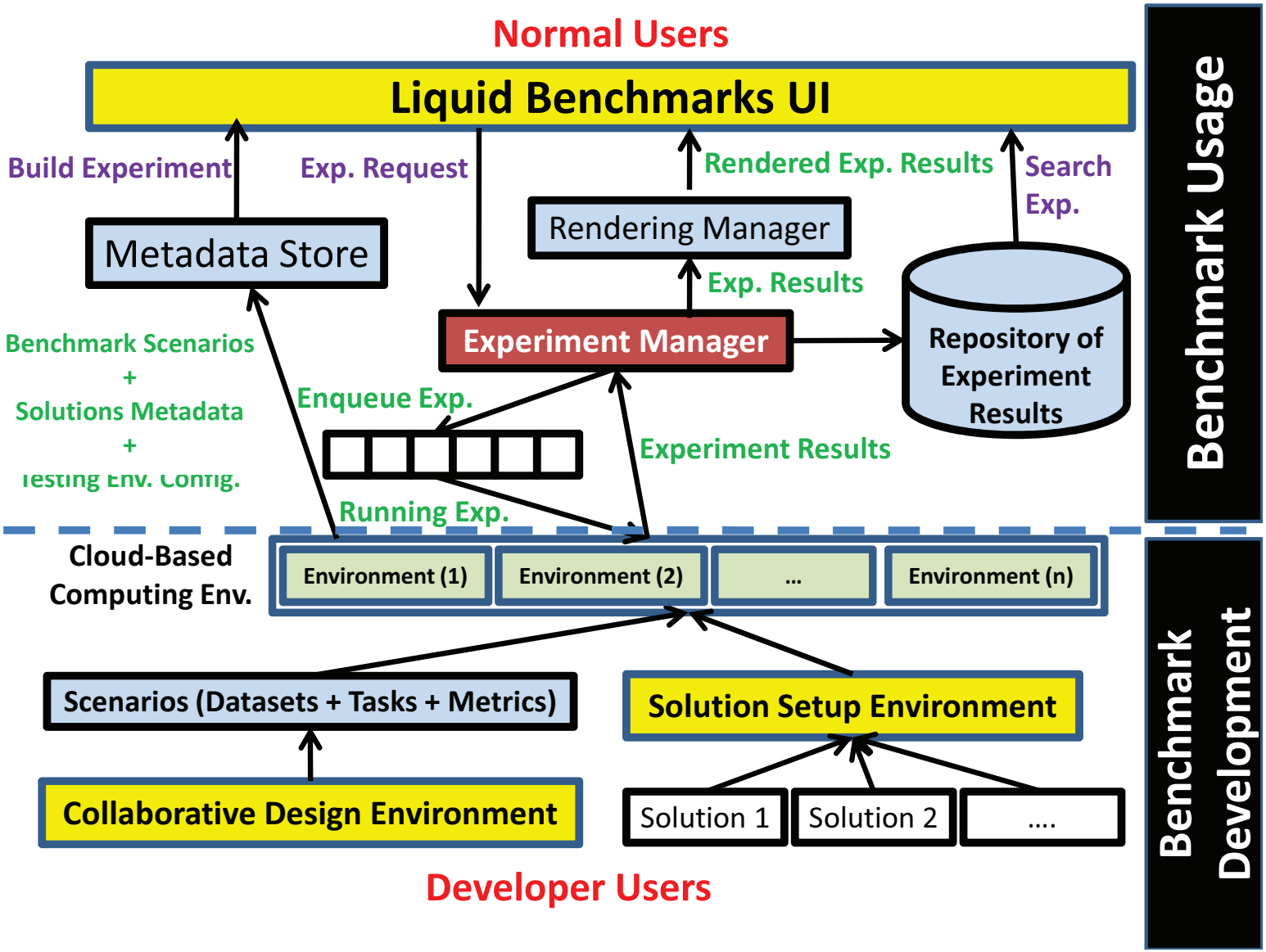
Continuous evolution of the state-of-the-art

- Experimental evaluation papers suffer from a main problem is that they represent *snapshots* for the state-of-the-art at the time of their preparation.
- By default, the research contributions in any field are always *dynamic* and *evolving* (e.g. new approaches, improvement for existing approaches).
- Experimental papers can go *out-of-date* after relatively short time of their appearance.
- Continuous maintenance of the published results may require too much work from their authors who may loose the interest to redo the job after sometime.

Liquid Benchmarks: Underlying Technologies

- **Cloud Computing**: as an efficient way of broad sharing of computer software and hardware resources via the Internet in an elastic way.
- **Software As A Service (SAAS)**: it provides the facility that *each* end-user does not require to manually download, install, configure, run or use the software applications on their own computing environments.
- **Collaborative and Social Web**: (e.g. Wikis, blogs, forums) offer a great flexibility in the ability of building online communities between groups of people that share the same interests (peers) where they can interact and work together in an effective and productive way.

Liquid Benchmarks: Architecture



Liquid Benchmarks: Components

- **Web-based User Interface**: design experiments, submit requests and search results.
- **Experiment Manager**: control the execution, ensure absence of influences. Receives, stores and renders the experimental results.
- **Repository of Experiment Results**: stores the results of all running experiments with their associated configuration parameters, provenance information (e.g. timestamp, user) and social information (e.g. comments, discussions).

Liquid Benchmarks: Components

- **Cloud-Based Computing Environments**: It hosts testing environments which are shared by the liquid benchmark end-users.
- **Collaborative Design Environment**: It is used to build the specification of the benchmark scenarios and provides the tools to achieve the task collaboratively (e.g. forums, wikis).
- **Solution Setup Environment**: It is used to setup and configure the competing solutions in the different testing environments (SAAS).

Liquid Benchmarks: Ongoing Case Studies

- XML Compressors.
- SPARQL query processors.
- Graph query processors.

Liquid Benchmarks: Benefits

- Providing **workable** environments to collaboratively build standard benchmarks.
- Developing centralized and **focused repositories** for related software implementations and their experimental results. That can be used as a very positive step to solve the **repeatability** problems.
- Facilitating **collaborative maintenance** of experimental studies to guarantee their freshness.
- Facilitate establishing **shared computing resources environment** that can be utilized by different active contributors in the same domain residing at different parts of the world.
- Leveraging the **wisdom of the crowd** in providing feedbacks over the experimental results in a way that can give useful insights for solving further problems and improving the state-of-the-art.
- Establishing a **transparent** platform for scientific **crediting** process based on collaborative community work.

Conclusions

- **Liquid Benchmarks**: A Step Towards An Online Platform for Collaborative Assessment of Scientific Research Results.
- We believe that the Computer Science research community should have the leadership to significantly improve the ability of assessing the impact of scientific research results.
- This work is at a preliminary stage and may leave out some of the important details (e.g privacy, credit attribution). However, we hope that our proposal will serve as the foundation of a fundamental rethinking of the experimental evaluation process in the computer science field.

End

- Please follow the updates of our project on

  http://project.liquidpub.org/research-areas/liquid-benchmarks

- Please email questions to: **ssakr@cse.unsw.edu.au**

THANK YOU