SIGMOD Officers, Committees, and Awardees

Chair	Vice-Chair	Secretary/Treasurer
Yannis Ioannidis University of Athens Department of Informatics Panepistimioupolis, Informatics Bldg 157 84 Ilissia, Athens HELLAS +30 210 727 5224 <yannis AT di.uoa.gr>	Christian S. Jensen Department of Computer Science Aarhus University Åbogade 34 DK-8200 Århus N DENMARK +45 99 40 89 00 <csj AT cs.aau.dk >	Alexandros Labrinidis Department of Computer Science University of Pittsburgh Pittsburgh, PA 15260-9161 PA 15260-9161 USA +1 412 624 8843 <labrinid AT cs.pitt.edu>

SIGMOD Executive Committee:
Sihem Amer-Yahia, Curtis Dyreson, Christian S. Jensen, Yannis Ioannidis, Alexandros Labrinidis, Maurizio Lenzerini, Ioana Manolescu, Lisa Singh, Raghu Ramakrishnan, and Jeffrey Xu Yu.

Advisory Board:
Raghu Ramakrishnan (Chair), Yahoo! Research, <First8CharsOfLastName AT yahoo-inc.com>, Amr El Abbadi, Serge Abiteboul, Rakesh Agrawal, Anastasia Ailamaki, Ricardo Baeza-Yates, Phil Bernstein, Elisa Bertino, Mike Carey, Surajit Chaudhuri, Christos Faloutsos, Alon Halevy, Joe Hellerstein, Masaru Kitsuregawa, Donald Kossmann, Renée Miller, C. Mohan, Beng-Chin Ooi, Meral Ozsoyoglu, Sunita Sarawagi, Min Wang, and Gerhard Weikum.

Information Director, SIGMOD DiSC and SIGMOD Anthology Editor:
Curtis Dyreson, Washington State University, <cdyreson AT eecs.wsu.edu>

Associate Information Directors:
Denilson Barbosa, Ugur Cetintemel, Manfred Jeusfeld, Georgia Koutrika, Alexandros Labrinidis, Michael Ley, Wim Martens, Rachel Pottinger, Altigran Soares da Silva, and Jun Yang.

SIGMOD Record Editor:
Ioana Manolescu, INRIA Saclay, <ioana.manolescu AT inria.fr>

SIGMOD Record Associate Editors:
Magdalena Balazinska, Denilson Barbosa, Chee Yong Chan, Ugur Çetintemel, Brian Cooper, Cesar Galindo-Legaria, Leonid Libkin, and Marianne Winslett.

SIGMOD Conference Coordinator:
Sihem Amer-Yahia, Qatar Computing Research Institute, <sihemameryahia AT acm.org>

PODS Executive: Maurizio Lenzerini (Chair), University of Roma 1, <lenzerini AT dis.uniroma1.it>, Michael Benedikt, Phokion G. Kolaitis, Leonid Libkin, Jan Paradaens and Thomas Schwentick.

Sister Society Liaisons:
Raghu Ramakhrishnan (SIGKDD), Yannis Ioannidis (EDBT Endowment).

Awards Committee:
Laura Haas (Chair), IBM Almaden Research Center, <laura AT almaden.ibm.com>, Rakesh Agrawal, Peter Buneman, and Masaru Kitsuregawa.

Jim Gray Doctoral Dissertation Award Committee:
Johannes Gehrke (Co-chair), Cornell Univ.; Beng Chin Ooi (Co-chair), National Univ. of Singapore, Alfons Kemper, Hank Korth, Alberto Laender, Boon Thau Loo, Timos Sellis, and Kyu-Young Whang.

[Last updated : June 15th, 2011]

SIGMOD Officers, Committees, and Awardees (continued)

SIGMOD Edgar F. Codd Innovations Award

For innovative and highly significant contributions of enduring value to the development, understanding, or use of database systems and databases. Until 2003, this award was known as the "SIGMOD Innovations Award." In 2004, SIGMOD, with the unanimous approval of ACM Council, decided to rename the award to honor Dr. E.F. (Ted) Codd (1923 - 2003) who invented the relational data model and was responsible for the significant development of the database field as a scientific discipline. Recipients of the award are the following:

Michael Stonebraker (1992)
Jim Gray (1993)
Philip Bernstein (1994)
David DeWitt (1995)
C. Mohan (1996)
David Maier (1997)
Serge Abiteboul (1998)
Hector Garcia-Molina (1999)
Rakesh Agrawal (2000)
Rudolf Bayer (2001)
Patricia Selinger (2002)
Don Chamberlin (2003)
Ronald Fagin (2004)
Michael Carey (2005)
Jeffrey D. Ullman (2006)
Jennifer Widom (2007)
Moshe Y. Vardi (2008)
Masaru Kitsuregawa (2009)
Umeshwar Dayal (2010)
Surajit Chaudhuri (2011)

SIGMOD Contributions Award

For significant contributions to the field of database systems through research funding, education, and professional services. Recipients of the award are the following:

Maria Zemankova (1992)
Gio Wiederhold (1995)
Yahiko Kambayashi (1995)
Jeffrey Ullman (1996)
Avi Silberschatz (1997)
Won Kim (1998)
Raghu Ramakrishnan (1999)
Michael Carey (2000)
Laura Haas (2000)
Daniel Rosenkrantz (2001)
Richard Snodgrass (2002)
Michael Ley (2003)
Surajit Chaudhuri (2004)
Hongjun Lu (2005)
Tamer Özsu (2006)
Hans-Jörg Schek (2007)
Klaus R. Dittrich (2008)
Beng Chin Ooi (2009)
David Lomet (2010)
Gerhard Weikum (2011)

SIGMOD Jim Gray Doctoral Dissertation Award

SIGMOD has established the annual SIGMOD Jim Gray Doctoral Dissertation Award to *recognize excellent research by doctoral candidates in the database field.* This award, which was previously known as the SIGMOD Doctoral Dissertation Award, was renamed in 2008 with the unanimous approval of ACM Council in honor of Dr. Jim Gray. Recipients of the award are the following:

- **2006** *Winner*: Gerome Miklau, University of Washington. *Runners-up*: Marcelo Arenas, University of Toronto; Yanlei Diao, University of California at Berkeley.
- **2007** *Winner*: Boon Thau Loo, University of California at Berkeley. *Honorable Mentions*: Xifeng Yan, University of Indiana at Urbana Champaign; Martin Theobald, Saarland University
- **2008** *Winner*: Ariel Fuxman, University of Toronto. *Honorable Mentions*: Cong Yu, University of Michigan; Nilesh Dalvi, University of Washington.
- **2009** *Winner*: Daniel Abadi, MIT. *Honorable Mentions*: Bee-Chung Chen, University of Wisconsin at Madison; Ashwin Machanavajjhala, Cornell University.
- **2010** *Winner:* Christopher Ré, University of Washington. *Honorable Mentions*: Soumyadeb Mitra, University of Illinois, Urbana-Champaign; Fabian Suchanek, Max-Planck Institute for Informatics.
- **2011** *Winner*: Stratos Idreos, Centrum Wiskunde & Informatica. *Honorable Mentions*: Todd Green, University of Pennsylvania; Karl Schnaitter, University of California in Santa Cruz.

A complete listing of all SIGMOD Awards is available at: **http://www.sigmod.org/awards/**

[Last updated : June 15th, 2011]

Editor’s Notes

Welcome to the June 2011 issue of the ACM SIGMOD Record, with many interesting articles and a set of news on the Editorial Board side!

The vision paper by Cohen-Boulakia and Leser opens the issue, with a perspective on how users devise, use, search for and interact with scientific workflows. Such processes have become ubiquitous as data produced and consumed by scientific activity increases in volume. However, even as specific tools become gradually known and used within scientific communities, the burden on the user trying to specify a workflow for a specific problem remains high: getting familiar with the workflow tool, the semantic of individual components, figuring out the overall interconnections between modules… The authors make a convincing case that the main way to ease this burden on the user is to improve/add to the ways in which a user may search for existing workflows, incrementally change or adapt them to her own problem. They survey the capabilities of various tools to search for processes matching a specification, and propose avenue for effectively encouraging scientific workflow re-use.

The survey by Wang, Feng, Xue and Song focuses on an important area of “green computing”, namely energy-efficient computing, and more specifically, energy-efficient data management. Energy efficiency is defined as the ratio between the work done, and the power consumed by the computing center doing the work. Several directions in the general area of energy efficiency are listed, such as system simplification (eliminating non-essential components), constructing energy optimizers and reducing peak consumption. With respect to the sub-topic of energy-efficient data management, the authors consider server-side, sensor-side and mobile-context techniques for increasing energy efficiency. The survey ends with a review of some metrics and benchmarks for quantifying this important measure.

The Distinguished Profile column features two interviews in this issue: Laura Haas and Alberto Laender, both reflecting on their rich and interesting careers. It turns out Laura had left school with no intention to work in databases, but the database community was lucky enough to have her work on a series of outstanding IBM projects which have opened fields of work such as data integration (Garlic), or schema mapping (Clio). Laura’s musings on management versus child raising will delight all members of our community which happen to be both parents and managers, and especially mothers! Besides his long and successful University professor career, Alberto has also created and sold an extremely successful Brazilian start-up to Google, and he explains the unusual circumstances which, together with the good timing and well-defined focus, lead to their success. Alberto also comments on old and new ways of assessing database (and more generally, computer science) research impact, on the structuring of the Brazilian database community around the SBBD conference, and much more.

The report by Meng, Jin, Cao and Yue outlines the works of the First International Workshop of Flash-Based Systems (FlashDB), held in conjunction with DASFAA 2011. The workshop focused on issues such as transaction processing, energy consumption, indexing, data placement etc. in the context of software systems designed for Flash hardware.

Finally, the report by Bonnet, Manegold et al. describes the results of the 2011 SIGMOD Repeatability and Workability evaluation. In 2011, this evaluation is in its fourth year; the authors report encouraging, yet stable results, and uncover some interesting and unexpected geographic trends in the repeatability results, for which everyone’s hypothesis are welcome…

New editors I am happy to announce a set of new arrivals in our editorial board.

The increased output of this month’s Distinguished Profile column is partly due to the presence of Vanessa Braganholo from Fluminense Federal University, our new co-editor of the column. Welcome, Vanessa!

After twelve years of service, Leonid is planning his withdrawal from the role of Database Principles editor. Pablo Barceló from the Universidad de Chile is joining us as a new co-editor, to take over from Leonid after a short transition. Welcome, Pablo!

Last but not least, after a period of inactivity, the Industrial Perspective column is re-launched by the arrival of a new editor: Glenn Pauley from Sybase, whose own short editorial appears below. Welcome, Glenn!

Your contributions to the Record are welcome via the RECESS submission site (http://db.cs.pitt.edu/recess). Prior to submitting, be sure to peruse the Editorial Policy on the SIGMOD Record's Web site (http://www.sigmod.org/publications/sigmod-record/sigmod-record-editorial-policy).

Ioana Manolescu

August 2011

From the new editor of the SIGMOD Industrial Perspective column

A tremendous asset to the database systems research community is the symbiotic relationship between academia and industry. Historically, that collaboration has resulted in a myriad of joint research projects, research prototypes, and commercial successes. Evidence of that relationship continues today, particularly so with the recent spawning of numerous commercial initiatives such as Hadapt, VoltDB, CouchDB, Voldemort, and MongoDB, to name but a few.

The intent behind SIGMOD Record's Industry Perspectives section is to inform researchers regarding industry trends, commercial product implementations, and the broader practice of data management, so that the database research community can better formulate research problems of practical significance. Topics of interest include, but are not limited to:

- surveys of data management practice;
- experience papers describing lessons learned from applying research results in an industrial context;
- data management standards;
- technical papers on commercial data management products, interfaces, and query languages;
- papers describing the requirements of novel applications;
- position papers; and
- papers from industrial researchers on their views of important research problems.

As the new SIGMOD Record Associate Editor for Industry Perspectives, I hope to follow Len Seligman's efforts as Associate Editor over the previous decade, and continue to contribute to the dialog between industrial and academic members of the database research community.

We encourage paper submissions on any topic along these lines. Industry Perspectives will be an ongoing feature of SIGMOD Record, and we welcome submissions at any time. Submissions should follow the formatting guidelines of SIGMOD Record and should be submitted electronically, in PDF format, to the RECESS system. Contributing authors maintain their copyright to articles published in SIGMOD Record. Paper proposals, inquiries, submission ideas, and topic suggestions for the Industry Perspectives section can be sent to Glenn Paulley at paulley@sybase.com.

Glenn Paulley
Sybase, an SAP Company
paulley@sybase.com

Past SIGMOD Record Editors:

Harrison R. Morse (1969)
Daniel O'Connell (1971 – 1973)
Randall Rustin (1975)
Thomas J. Cook (1981 – 1983)
Jon D. Clark (1984 – 1985)
Margaret H. Dunham (1986 – 1988)
Arie Segev (1989 – 1995)
Jennifer Widom (1995 – 1996)
Michael Franklin (1996 – 2000)
Ling Liu (2000 – 2004)
Mario Nascimento (2005 – 2007)
Alexandros Labrinidis (2007 – 2009)

Search, Adapt, and Reuse:
The Future of Scientific Workflows

Sarah Cohen-Boulakia
Université Paris-Sud 11 CNRS
UMR 8623, AMIB INRIA Saclay
cohen@lri.fr

Ulf Leser
Humboldt-Universität zu Berlin
Unter den Linden 6, 10099 Berlin
leser@informatik.hu-berlin.de

ABSTRACT

Over the last years, a number of scientific workflow management systems (SciWFM) have been brought to a state of maturity that should permit their usage in a production-style environment. This is especially true for the Life Sciences, but SciWFM also attract considerable attention in fields like geophysics or climate research. These developments, accompanied by the growing availability of analytical tools wrapped as (web) services, were driven by a series of very interesting promises: End users will be empowered to develop their own pipelines; reuse of services will be enhanced by easier integration into custom workflows; time necessary for developing analysis pipelines will decrease; etc. But despite all efforts, SciWFM have not yet found widespread acceptance in their intended audience. In this paper, we argue that a wider adoption of SciWFM will only be achieved if the focus of research and development is shifted from methods for developing and running workflows to searching, adapting, and reusing existing workflows. Only by this shift can SciWFM outreach to the mass of domain scientists actually performing scientific analysis – and with little interest in developing them themselves. To this end, SciWFM need to be combined with community-wide workflow repositories allowing users to find solutions for their scientific needs (coded as a workflow). In this vision paper, we show how and where such developments have already started and highlight new research questions arising.

Keywords

Scientific Workflow Systems, Workflow Management, Scientific data, Data Analysis.

1. INTRODUCTION

In the last decade, considerable effort has been put into the development of scientific workflow management systems. These systems primarily aim at supporting domain scientists in developing, running, and monitoring data analysis programs coded as workflows [DGST09]. A variety of systems, including Taverna [OGA+05], Kepler [LAB+05], Pegasus [DSS+04], VisTrails [SVK+08], and Triana [CGH+06], have reached a level of maturity that in principle allows them to be used by scientists for their daily needs. The goal of these systems is to put away the disadvantages of the state-of-the-art in developing scientific analysis, which is implementing custom programs, mostly in Perl or similar scripting languages. SciWFM promise (to varying degrees) to enable development of analysis pipelines at a higher level of abstraction, to take care of logging, provenance management, process control, recovery, scheduling and parallelization of individual tasks, and to increase understandability and sharing of workflows. SciWFM also could be a key infrastructure for repeatable science [GNT+10].

Until now, however, uptake of SciWFM has been limited. A search for "scientific workflow management" in Google Scholar lists more than 1,000 papers. However, many of them are concerned with research in SciWFM involving Computer Scientists, and not with their use by scientists. For instance in the Life Sciences, one of the major focuses of papers on SciWFM, we are aware of only a few papers that report on using a SciWFM for science, many of which were co-authored by developers of SciWFM systems. Though it might be that more users use SciWFM without mentioning it in their publications, we believe that it is safe to state that SciWFM have not yet reached their intended user group to a satisfying level.

There are many speculations on reasons for this situation. We believe that, actually, SciWFM have yet failed to properly define and target their intended user group. For a domain scientist, today's systems are much too complex. In the same manner as such users don't want to write Perl scripts or SQL queries, they don't want to model and program analysis workflows. Furthermore, for these people a workflow is not easier to understand than a program; on the other hand, these are exactly the people that have regular and pressing needs for performing data analysis on the masses of new data they generate daily [AKD10]. The competitors for SciWFM with respect to this user group are packed (and potentially commercial) applications (in Transcriptomics, from which we will draw our examples in the rest of this papers, this would be systems like Chipster or GeneSpring). For a Bioinformatician, on the other hand, SciWFM are too cumbersome and inflexible. These users are mostly interested in developing new methods and are usually capable of writing their own analysis scripts [HMB07]. Today they can take advantage of the large number of freely available libraries (such as BioPerl, BioJava, BioSQL, or R) for performing standard tasks, which already let them concentrate much more on their problem at hand than it was possible 10 years ago. A SciWFM offers them few advantages over scripting.

Furthermore, SciWFM have some inherent properties diminishing their potential advantages. First, a complex workflow composed of dozens of intertwined tasks, in general, is not much easier to understand than a well structured program performing the same analysis. Second, the ability to seamlessly integrate external services often leads to bad performance (as large amounts of data need to be passed around in wide-area networks), turns debugging into a nightmare, and also makes workflow execution dependent on the reliability of these services. Finally, experiences show that services are easily listed in a repository, but that they do not easily work together due to incompatibilities of data types, formats, invocation methods etc. Instead of concentrating on the scientific tasks, developers of scientific workflows therefore still have to spend a big deal of their time in writing “glue” code [RLS+06].

These observations call for a fresh view on SciWFM. We argue that SciWFM will have a hard time in persuading developers of new analysis methods to use them (called “power users” from now on). At the same time, by focusing exactly on this class of people, the SciWFM community ignores the much larger group of users that actually could benefit the most of SciWFM, i.e., the domain scientists (“true users” from now on) [Ste08]. These people do not know how to program, and they don’t know how to use a SciWFM to develop a workflow – and they probably never will. However, true users are in urgent need for new methods to analyze their ever growing flood of data. They are interested in applying advanced methods (which they don’t even need to understand) on new data to discover new facts in their respective science. Essentially, they are searching a solution to a given problem instead of aiming at developing a solution themselves. This currently bounds them to packaged software, often commercial, which ties them to the vendor / creator and cuts them off from using the most recent methods. At the same time, they would benefit tremendously from all the runtime-support that a SciWFM offers – but the technical obstacles currently are too severe for them to be able to take advantage.

In this paper, we provide a vision on what we believe could be the killer application for SciWFM – focusing on true users. Essentially, we take a new point-of-view on an existing class of systems which would make them attractive for masses of new users. We describe use cases that would be possible when following our novel point-of-view, show existing developments in this direction, and describe technical challenges arising from our vision. Adequately supporting true users requires a couple of things, such as domain-specific repositories of existing workflows, methods to find workflows solving a given analysis problem, and algorithms for comparing different workflows solving the same problem. Taken to the extreme, in our vision SciWFM will more and more become a largely invisible part of the computational infrastructure of a scientist, such as a database or an operating system is, that most (true) users do not directly deal with anymore. Instead, they simply chose a workflow, provide their data, maybe set some parameters, and press the “run” button. All the technology for scheduling, monitoring, restarting, service invocation, format conversion etc. remains hidden.

Notably, we do not believe that power users should or will stop using SciWFM. These people, though not in numbers, will continue to use SciWFM for developing new methods. Actually, if the true users would start to use SciWFM more often, this could also increase the attractiveness of SciWFM for power users, as publishing their workflows in such a system would increase the chances for them of being reused.

2. State-of-the-Art in SciWFM

This section gives an overview over the state-of-the-art of current SciWFM systems. We focus on those concepts that are critical for the support of “true users”. For a general review on SciWFM see [DGST09].

2.1 Workflow Design

One of the most important claims of SciWFM it to provide an environment to ease the design of scientific analysis processes. The claim is that by representing these processes at a higher level of abstraction, their understandability, reuse and modularity would be enhanced [LAB+05]. Along this line, all SciWFM are equipped with a GUI in which users may drag and drop boxes (tasks) and connect them to describe data or control flow which should raise the general level of feeling “it is simple”.

However, while domain scientists usually have an idea of the kind of analysis they want to perform on their particular data set, they often have no detailed knowledge of which concrete tools to use (and the parameters to set) and how to combine them into a complete workflow that would execute their intention. Several proposals have appeared to help in this respect. First, repositories of tools and services have emerged, such as BioCatalogue listing more than 3,000 web services [BTN+10] indexed by keywords and ontology terms. Second, tools have been developed that allow automatic translation of abstract specifications into concrete workflows [GGW+09; LAG03]. However, we have doubts whether such attempts are particularly helpful for the true user. Large repositories are impressive, but searching them is hard. True users are rather overstrained than supported by the amount of services being offered [DRGS09]. They are often not capable of (or not willing to) taking a rational choice. Furthermore, users still have the major problem of concretely chaining the tools together to solve their problem. On the other hand, systems supporting workflow inference require a level of detail in description of the available services that may render them inapplicable in any larger scenario.

2.2 Models of Workflows and Workflow Runs

In SciWFM, one has to distinguish between a workflow specification, given in some kind of modeling language, and the concrete workflow runs (or traces), which are series of executed tasks that conform to a workflow specification. Both aspects are treated very differently in different systems.

1. Currently, essentially every SciWFM has their particular language for specifying a workflow, equipped with a particular semantics of how such a specification is interpreted as an executable program [MLB+10]. Things get even more complicated with systems that allow dynamic adaptation of the semantics of a specification; for instance, the Kepler system offers different so-called directors for orchestrating workflow execution, leading to entirely different execution threads [DKM+05].
2. Storing, modeling and searching runs recently have become prominent research topics, usually summarized under the term “provenance management”. Provenance is a key concept for SciWFM since it supports reproducibility of how results were obtained and helps assessing their quality [BCB+08; DF08]. Approaches for managing provenance differ greatly in various aspects, such as the degree of granularity or the models used for representing runs [CBT09; MLA+08]. Further, the concrete relationship between a workflow and its trace varies a lot due to the differences in the semantics of workflow specifications.

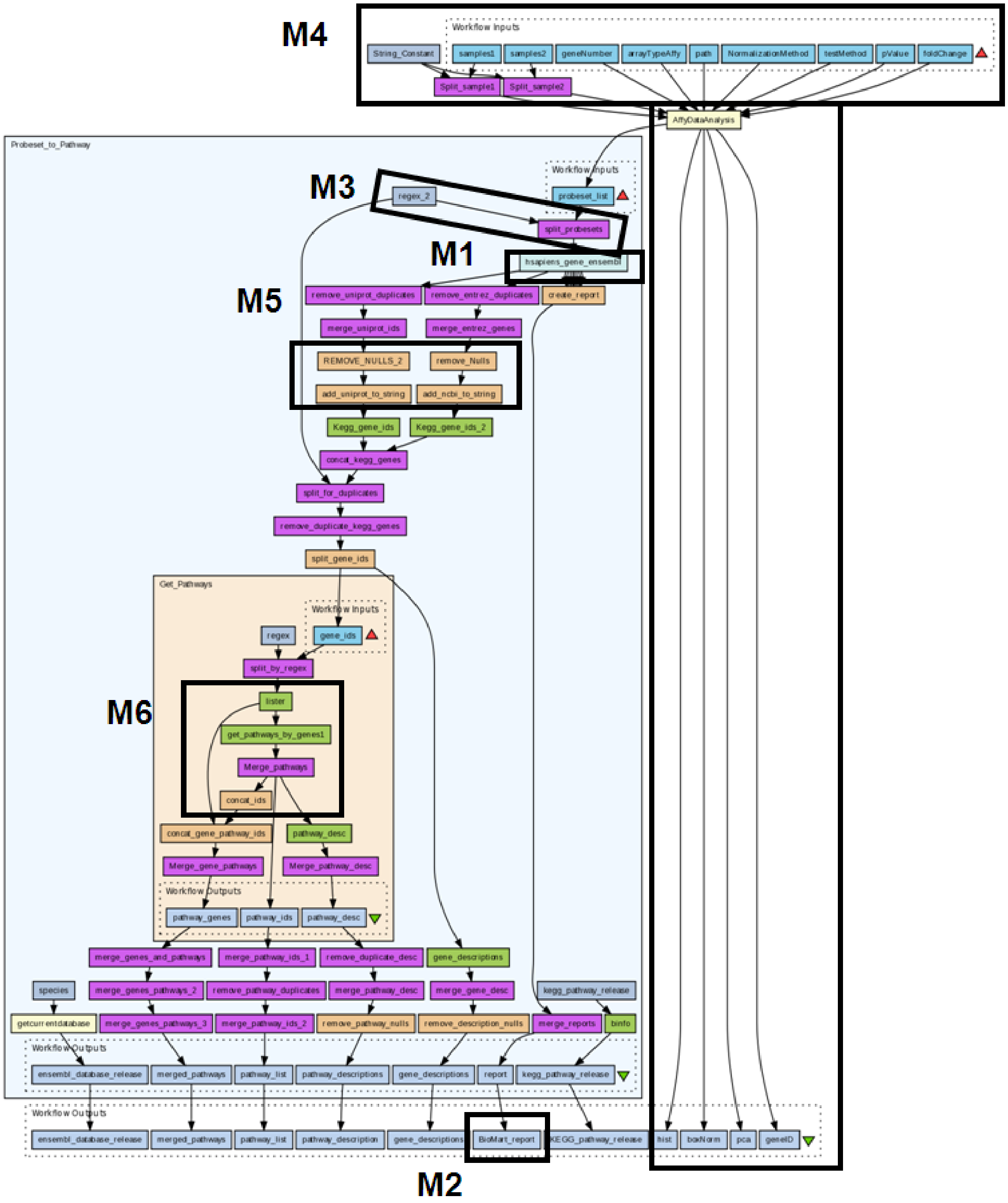


Figure 1 : One Workflow from myExperiment (W10) composed of 80 processes including 9 inputs and 11 outputs and processes of various kinds (e.g., green processes for calls to Web services, purple for local Java code) and including two sub-workflows (denoted by larger boxes in light blue and light orange) to enhance modularity and reusability.

From a user perspective, this heterogeneity is a nuisance. Although there are movements towards unifying provenance management [MLA+08], the current state-of-the-art is characterized by a high level of heterogeneity between systems, both on the trace and on the specification language. As a consequence, a true user has no way of comparing the specifications of two workflows from two given systems, nor is it possible to port a workflow from a system X to a system Y. Users are therefore bound to one specific SciWFM, which severely restricts the choice of solutions they could search in theoretically.

2.3 Workflow Repositories

Clearly, our vision of true users searching the solution to their problem among existing workflows critically depends on the existence of large repositories for workflows supporting powerful methods of search and reuse. Some such systems have already been developed, offering for the first time to the community a plethora of real examples of scientific workflows. The precursor and major project in the Life Sciences is myExperiment [DRGS09] (http://www.myexperiment.org/), a web-based public repository of workflow definitions with the explicit aim to enable sharing of those. As of June 2011, more than 1,300 workflows are freely available. Most of the workflows in the repository are from Taverna, but myExperiment also hosts some workflows from other systems. Any scientist can upload workflows in my Experiment. The site also supports the typical Web2.0-style features like registration, group-building, and biding / commenting on workflows. myExperiment already was the subject of user evaluations, though only to a limited scale and with no clear results [GDR+08].

Systems such as myExperiment are at the basis of our vision, but we strongly believe that major features and functionalities must be added to be truly useful to end-users. First, the rich structure of workflows should be considered as first-class citizens in these systems while they are currently considered only as binary black boxes with annotations. Second, searching for sub-workflows, searching based on the features of runs previously executed, sensible ranking of results, or workflow comparison (see Chapter 3 and 4 for details) should be provided. Currently, searching workflows is only possible by keywords; the repository does not offer advanced similarity search methods that could, for instance, take the topology of workflows into account (e.g. when the user wants to describe the sequence of steps to be performed).

CrowdLabs is another repository launched recently; although it is not dedicated to a particular kind of workflow, it currently only stores workflows concerned with visualization designed using the VisTrails. As for now, its community of users is smaller than the community of myExperiment. Kepler supports the installation of local repositories but offers only basic search capabilities (keyword-based search) [B. Ludäscher, personal communication]. Commercial systems like InforSense or Pipeline Pilot also offer this feature. Interestingly, we also are not aware of any comprehensive attempts in public repository building in the closely related fields of business workflows or ETL (see [Alb09; GDR+08]). This probably can be attributed to the fact that business workflows (as ETL processes) are rarely shared between communities; instead, the set of workflows of a company are a major commercial asset that is intensively curated and kept secret. Accordingly, workflow sharing has not achieved the level of importance as in an open domain such as Bioinformatics, where the exchange and publication of new methods has a long tradition [GS08].

3. Supporting True Users

The current state-of-the-art in SciWFM is largely driven by attempts to ease the development and execution of workflows. In this paper we argue that it is time to change the focus and concentrate on supporting people that do not want to develop workflows anew, but that instead are interested in getting their data analyzed with as little effort as possible and with the best methods that exist, i.e., the “true users”. In this section, we describe the kind of problems those people have when they try to use a SciWFM for this purpose. The different use cases will be broken down to specific technical challenges in the next chapter. All use cases we describe are inspired by concrete analysis tasks we have encountered in various multidisciplinary Life Science projects. In particular, we will consider one main domain of application within the Life Sciences, namely, microarray data analysis. For an introduction into this field, see, for instance, [CH07; LKMS08].

We base our examples on real workflows extracted from myExperiment. Table 1 gives an overview over the selected workflows (names, main inputs and outputs, number of processes composing each workflow, sub-workflows involved, and species analyzed). Note that these workflows, as are the underlying analysis problems, are fairly complex and far from the toy examples usually discussed in the SciWFM literature. For illustration, Figure 1 shows W10 as a screenshot from the Taverna workbench.

3.1 Reusing Existing Workflows

The fact that the Life Sciences are a field in which new discoveries are achieved at an impressive speed does not imply that each discovery would have a new method accompanied. Instead, new methods are rather rare; once established, they are typically used for thousands of cases. For instance, modern gene microarrays allow measuring the expression levels of all transcripts in a given sample at once. This can, for instance, help to find genes behaving particularly under certain circumstances. Since the development of this biotechnology method in the mid 90ties, it has been applied in hundreds of labs world-wide and is an offer of many commercial biotech companies. Probably any disease by now has been studied using microarrays, and most of them multiple times [LKMS08]. The methods used in these experiments are not identical due to variations in the chips being used or the question being studied, but in general they are highly similar. For a given chip type, de-facto standards exist, though they may change over time [HSRC08].

True users of gene expression data typically focus on producing large amounts of high-quality experimental data. Having the data, they want them to be analyzed using the best possible methods. Such users would be highly interested in searching a repository of workflows for analyzing microarray data. The choice of a particular workflow would be driven by the amount of data produced, the concrete chip type that was used, and the primary biological question one has in mind. It would be the task of the repository to find all workflows matching the expressed constraints and rank them according to some criteria, such as popularity, reliability or physical location of services being used. The user would then simply choose one of the matching workflows, provide its input data, and click the “run” button. A local SciWFM should take the data and download and execute the chosen workflow, tracing all steps in a provenance model that may also be uploaded to the repository for later reuse (see below).

ID	Workflow Name	Input	Output	Species	#Proc	Sub-workflows
W10	Human Microarray CEL file to candidate pathways	Affymetrix: .cel file + normalisation method	List of the top differentially expressed genes (info on pathways and annotations on genes)	Human	80	Get_pathways, probset_to_pathways
W19	Mouse Microarray Analysis	List of Probe sets (from Affymetrix microarray)	Mapped Ids (Kegg, uniprot, ensembl)	Mouse	54	Get_pathways
W40	Microarray CEL file to candidate pathways	Affymetrix: .cel file + normalisation method	Diff. expressed genes, pathways, gene annotation, BioMart report	Human	81	Get_pathways, CandidatePathways
W79	Mapping microarrays onto pathways	sbml	image sbml model processes	*undefined*	41	writeSBML Extractgene
W142	Microarray CEL file to candidate pathways	Affymetrix: .cel file + normalisation method	Differentially expressed genes, pathways, gene annotations	Mouse	81	Get_pathways probeset_to_pw
W143	Human Microarray Analysis	List of Probe sets (from Affymetrix microarray)	Mapped IDs (Kegg, uniprot, ensembl)	Human	54	Get_pathways
W187	From cDNA Microarrays to Pathways and Abstracts	cDNA	Differentially expressed genes, pathways, pubmed report	E. coli	73	Search pubmed, retrieve abstract
W174	AffyArrayQualityAnalysis	CEL files	Quality control scores	*undefined*	25	Check_status, download_files

Table 1. Overview over the eight workflows used as example in this paper. A workflow denoted as Wx is available at http://www.myexperiment.org/workflows/x.html, where x is the id indicated on Table 1. “#Proc” gives the number of individual processes a workflow is composed of. “Sub-workflows” lists those workflows that are embedded into the main workflow by name. Names sometimes are abbreviated to save space.

Illustration: The workflows of Table 1 are all related to microarray data analysis. Some but not all of them are dedicated to Affymetrix data analysis (a given brand of chip, see column “Input”). Such workflows may still differ in (1) the format expected for the input, conforming to different levels of pre-processing of the raw data, (2) the biological question they answer and thus the kind of output they provide (pathways, gene and/or protein annotations, information about the quality of the input data), and (3) the kind of individual services they use, which can range from sequences of external services to monolithic custom-developed scripts.

Supporting a user in finding the right workflow for his purpose in this setting requires first to enable her to clearly specify their needs. This can, for instance, use keyword-based searching, specification of type-constraints on input- or output, searching with a rough idea on the types of analysis steps performed, etc. We elaborate on these options in Section 4.1

3.2 Searching and Comparing Workflows

Having performed an analysis with a concrete workflow X, many users will be interested in also trying other pipelines for the same task. This can be achieved by either repeating the steps described in the previous section, or by letting the system search for another workflow that is similar to X. Thus, given a concrete workflow X, the repository should be able to search and rank workflows that are similar to X. This requires similarity measures adapted to the domain of SciWFM: They should take into account things such as input and output types of individual tasks and of the entire workflow, the individual purpose of the tasks involved, and the order in which the tasks are to be executed, i.e., the topology of the workflow graph.

Having selected a similar workflow Y, a user often will be interested in comparing X and Y. This, again, should be supported by the repository. The search algorithm thus must be able to not only measure an abstract similarity, but also to pinpoint the concrete differences between two workflows, which is a problem of workflow alignment. Another interesting functionality is to cluster all workflows resulting from a search into groups of highly similar specifications; this would enable the user to choose truly different methods for validation runs instead of repeating again and again an almost identical workflow. Furthermore, clustering the result space is very helpful for giving an overview over the range of solutions contained in a repository.

Illustration: Let us consider that our user is working on a given disease. Using results of microarray experiments, she is interested in biological pathways that behave specifically in the diseases samples. Suppose she has used W10 for this purpose and is now looking for alternatives. Based on the kind of input/output she provides and wants to obtain, the SciWFM could suggest workflows W40, W142, W19, or W143 as possible alternatives.

A deeper look at the set of workflows S1={W10, W142,W40} shows that those workflows are rather similar in terms of structure. Figure 1 represents W10 and we have underlined (globally) the portions of W10 which are modified in the workflows described hereafter. W10 and W142 only differ in the genome used but perform the exact same series of steps, except one formatting step (M1 is replaced by one alternative module) dedicated to the species. W40 differs in the output provided (M2 is missing) and performs formatting steps on the input in a different way (M3 is replaced by several other modules branched in another part of the workflow). As a consequence, the (names of the) sub-workflows used vary: probeset_to_pathways vs candidate_pathways. Accordingly, S1 should form a separate cluster of workflows in the search result and S2 = {W19, W143} should form the other cluster. Interestingly, all together, workflows from S1 and S2 are also both structurally and intentionally very similar. They diverge all on the first processes (roughly speaking, M4 is removed is replaced by a different suite of modules) because the data format is different (list of probe sets vs /cel file). Variation between S1 and S2 also occur at the level of modules M5 and M6. Thus, in a hierarchical clustering {S1}∪{S2} may form a bigger cluster on a higher level.

While topological aspects may be important to determine workflow similarity, it is not the only feature to be considered. W187 is, for instance, close to workflows from S1 and S2 in the sense that the input/output of the main tasks (from chip data to pathways) are the same. However, it is structurally very different. This is also true for W79 that, in contrast to the other workflows, invokes complex library functions (that remain black boxes to the workflow engine) instead of separate services for separate tasks as the other workflows do. This renders, for instance, W79 to be structurally very different from W142 although the achieved functionalities are very similar.

3.3 Adapting Existing Workflows

The former two use cases aimed at finding an existing workflow for analyzing a dataset produced with an established technology. However, sometimes users have more innovative datasets whose analysis requires the adaptation of existing workflows. For instance, there exist various technological variations of gene microarrays. One example are exon arrays, where the expression of single exons instead of entire genes is measured. Measuring exon arrays is a fairly new technology, and no standards for analyzing them have emerged, yet; however, many steps from gene expression analysis can be directly reused [ZL10].

Illustration: Essentially any of the workflows described in Table 1 can be adapted to the analysis of exon arrays, though it would require different amounts of work. In any case, the first steps (normalization,,mapping probesets to transcripts) need to be adapted, while all the downstream analysis (search for affected pathways, comparison with literature etc.) can be performed in exactly the same way as with classical microarray data. Depending on the workflow, changing the first steps can be done in a few minutes or may require more work, especially when it has been implemented using custom scripts. The smallest number of modification would be necessary for W79 which is built completely on packages from Bioconductor that can work both with gene and exon level measurements.

Faced with such a novel, yet not very distinct need, a user would first perform a search as in the previous examples. However, the repository would not be able to find a complete match and thus would inform the user that some constraints of the search could not be fulfilled, implying that the workflows returned only partly match the query. These results can be a starting point for the development of a new workflow by adapting one of the results. Workflow adaptation should be supported by the system, by clearly distinguishing matched parts of the query from others, and possibly by also suggesting solutions for the unmatched parts of the query by iteratively performing individual searches on those parts. Similarly, if all matches found are too distinct from the original needs, the user may chose to pose queries for parts of his intended analysis only.

Illustration: Consider the case where a user searches for workflows taking in genomic sequences from microarray data and performing normalization and functional pathway analysis. Though all workflow of S1 are candidates for the second part of the requirements, they do not accept sequences as input. They would thus have to be adapted by adding new tasks that extract sequence information from public databases and provide a new kind of input. As a solution for this partial requirement, the repository could suggest the sub-workflow extractgene of W79; the requirements of the user could thus be covered by chaining this sub-workflow with the “downstream” parts of any of the workflows from S1.

3.4 Assembling from Partial Solutions

If the amount of necessary adaptation exceeds a certain limit, the problem slowly becomes more and more similar to that of assembling a new workflow from partial solutions. This is the most complicated scenario occurring when small subparts of the intended analysis are already available in the repository. At the same time, this use case is, in the continuum from simply reusing a workflow to developing a new workflow, the closest to the latter and thus a task that is more typical for power users than for true users. One example from transcriptomics is the emerging trend to replace hybridization of probes to an array (as used in microarrays) with directly sequencing the expressed mRNA. A comprehensive analysis pipeline for such data could take building blocks from various existing workflows (sample preparation, data normalization, sequence-to-genome mapping, sequence assembly, copy number detection etc.), but the changes to a microarray workflow are so severe that the assembly would feel much like developing a new workflow.

An intelligent repository also can be of great help in this case. For instance, the assembly can be supported by checking format and type compatibility of the different parts, by suggesting glue-code for performing necessary data transformations, or by rewriting workflows into equivalent ones by identifying and removing redundant parts stemming from merging semantically overlapping workflows.

3.5 Exploiting Information about Runs

In many of the tasks described above, users may exploit information about runs of workflows that have been executed by others or by themselves and that are stored in the repository. First, this may allow users to find workflows that have been used on very similar data (e.g., gene expression from the same tissue or species). It may also be used to point a user to workflows used on data similar to their own input. For instance, some tasks may perform better or worse when used with eukaryote or prokaryote data, two families of species that are evolutionary very distant. Additionally, accessing previous runs may help to avoid rerunning subparts of workflows that have been run with the same data and parameters already. Such repetitions are common place in bioinformatics; however, supporting them would also require storing the results of workflow tasks.

Illustration: Consider that our user is looking for a workflow for analyzing microarray data from human Liver cells. This user would be highly interested to know on which tissue types the workflows meant for human data (W10, W40 and W143) have been run in the past, and how confident the users of these pipelines were with the results.

Another common scenario in bioinformatics is the repeated running of the same workflow on the same data, only applying small changes to some parameters. Even small variations often lead to very different final outputs. For example, the set of differentially expressed genes of the same microarray will not be the same if the search is performed with different thresholds for deciding which genes are considered as "differentially expressed". In this setting, users often would like to know why the results of two runs are different in the sense that they would like to see the point in the analysis where the intermediate results started to deviate. Finding this point can be achieved by comparing both traces. Such information is crucial to choose between two conflicting results or to assess the quality of outputs.

4. Research Directions

The use cases presented in the previous chapter can be broken down to a number of technical problems which need to be solved to make such modes of usage possible, i.e., to better support true users. In this chapter, we list those technical problems, discuss existing pieces of work targeting their solution, and point to open problems and novel research directions.

4.1 Describing what Users Search

Allowing users to express what they search is a key functionality of workflow repositories. Clearly, true users will not be experts in any query language and should be able to express their request as easily as possible. It would be highly advantageous if such queries, even if expressed verbosely in first place, would internally be rewritten into a formal query language; power users may be able to query the repository directly using it. Accordingly, we believe that a continuum of specification languages should be provided depending on the level of expertise in the user and the clearness of the requirements.

At one end of the spectrum are simple keyword queries that search a textual description of a workflow. Users describe their data and intended analysis by a set of keywords that are matched against the workflows, i.e., their documentation, metadata, data types, task names etc. This is the only type of search supported in current systems [OGA+05]. An improvement to this flat style of queries is to apply keyword searches on different levels of abstraction of a workflow, as recently proposed in [LSC10]; however, finding the right level of abstraction is a non trivial task in the complex and nested workflows typically used in science. On the other end of the spectrum, repositories should support full-fledged query languages encompassing predicates for searching IO-types of tasks, the topology of a workflow, keywords in the descriptions of tasks or the entire workflow, etc. A number of proposals in this direction have recently been put forward which we discuss in Section 4.2.

Largely unexplored are solutions in between. For instance, true users often know the type and format of data they provide (like Affymetrix GeneChip raw measurements, file format: CEL) and the type of data they expect as output (list of differentially expressed genes). Accordingly, they should be able to specify such parameters as constraints for a search, while the "intermediate" bits remain unspecified or only vaguely described. Often, true users also have a coarse-grained picture of the analysis they want to perform (like: "Data should be normalized, then aggregated by sample group, then analyzed for differential expression using a statistical test with a multiple testing correction"). The topology of the workflow can be (partly) given in such descriptions. Accordingly, queries should be augmentable with partial (high level) representations of the intended analysis. We call this a workflow sketch: representing some abstract tasks to be considered, their order, or even giving hard constraints for some tasks. Works conducted on the correspondence between abstract and concrete workflow specifications (e.g., [GGW+09; LAG03]; also see Section 2.1) may provide some hints into this direction; however, their overall intention is quite different from our vision. Another interesting starting point could be visual query languages [CM90; HS08]. Furthermore, a user should be able to express preferences about the kind of tools to be considered and the data sources to be used which could use techniques as those described in [BFL+04].

4.2 Searching Workflow Specifications

4.2.1 Searching with Workflow Sketches

While models for workflow sketches and their translation into a formal query are (to our knowledge) problems that have not been tackled before, there exist several works on query languages for workflow specifications. Most existing approaches were developed for business workflows, such as the Business Process Query Language (BPQL), BPMN-Q [AS10] or BP-QL [BEKM08]. Such approaches usually retrieve workflows based on the types of inputs and outputs of tasks. BP-QL [BEKM08] also allows the definition of flow patterns in the query while BPMN-Q enables predicates on the topology of a business workflow. Whether or not these approaches also are applicable to SciWFM is an open question.

However, none of the existing approaches allow users to express arbitrary constraints on the graph structure of the workflow, for instance to explicitly search for loops or branches. Additionally, all apply strict matching techniques and cannot rank results by similarity. Furthermore, searching heterogeneous workflows (workflows defined in different languages) is a seemingly unexplored area, though heterogeneous repositories already are a reality (such as myExperiment). Essentially, languages should be able to support arbitrary combinations of syntactic and semantic predicates and should be able to either apply strict or approximate matching.

4.2.2 Searching Similar Workflows

In the previous section, we described methods to search a workflow having only a rough idea about what it should do. But, as described in Section 3.2, it is also very interesting to search workflows similar to a concrete other workflow. However, if the specification language allows forming arbitrary graphs (especially containing loops) already searching only topological identical subgraphs is equivalent to the NP-complete subgraph isomorphism problem. However, it is not yet clear whether scientific workflow graphs really are arbitrary graphs, i.e., whether the description of scientific analysis pipelines really needs all possible ways to connect tasks. Interestingly, some workflow languages lead to graph structures that have a certain form, the most important example of which are series-parallel graphs (SP). [ZCBD+09] showed that the problem of matching SP graphs can be solved in polynomial time, which opens the door to a new class of efficient matching and ranking algorithms. Finding a similar characterization for other workflow languages is an open and important problem.

Finding similar workflows is directly associated to finding a proper similarity measure for workflows. Similarity measures for workflows should consider syntactic and semantic aspects, i.e., both the graph structure and the nature of the individual tasks (including names, input and output types, etc.). Proposals for the first problem are scarce. [GCB08] described a graph matching approach to match behavioral descriptions of services, but their method does not scale to large repositories and does not provide approximate matching. Other approaches exist in the graph search community but have not yet been applied to scientific workflows [HS08]. Ideally, when comparing structures, a rich set of edit operations (e.g., add/remove/replace a flow or a task) should be considered and weighted individually. Such weights could be derived from levels of confidence in tools assigned by a user or by experts. This allows distinguishing workflows logically performing the same tasks but using tools of various levels of reliability. From the semantic point of view, the similarity measure may exploit the fact that data types (at least in the life sciences where an increasing number of tools have their input/output described through ontologies) are readily organized into hierarchies. Approaches from the data mining community (e.g., [CBGZ06] considering distances between terms within a hierarchy for matching may be used in this context.

There is also ongoing work tackling the similarity of workflows from a radically different point of view. In these works, workflows are modeled as Petri Nets and are compared by their behavior under all possible inputs [RFL06]. Whether or not such methods are applicable for scientific workflows is not yet clear.

4.2.3 *Searching Sub-workflows*

The former methods always compare two workflows entirely. But sometimes it is also interesting to find a workflow which contains a sub-workflow that matches best to a sub-workflow of the query (also called local alignment instead of global alignment). To our knowledge, this problem has not been addressed for workflows so far. Methods can be inspired from previous work on local alignment of strings [Gus97] and, more interestingly, on schema matching in XML [ADMR05]. Approaches like BP-QL work on BPEL specifications represented as XML documents [BEKM08] but always consider the entire workflow and only perform exact searching.

4.2.4 *Clustering Workflows*

In large repositories, users could benefit greatly from an intuitive structuring of potentially large result lists. Apart from ranking according to the quality of a match, results should also be grouped into clusters of similar workflows. Depending on the similarity measure, workflows can be clustered such that each workflow in a cluster essentially solves the same type of problem but uses a different approach. Such a clustering can also help in browsing by automatically structuring the space of available workflows.

The technical challenge here is probably less novel than for the other problems we describe; however, we are not aware of any existing work in this direction. Finding a proper similarity measure is, again, the central problem. Furthermore, the clustering should be able to differentiate between hard constraints (like data types to work on, expected result) and soft constraints (like concrete implementation of a particular task, topology of the data flow). Techniques for clustering database schemas [SMH+10] or structured interfaces [WYDM04] may provide hints for this problem.

4.2.5 *Interactive Search*

A particularly important feature would be to provide an interactive search interface. Users being presented a list of matching workflows probably have a hard time to decide which one to choose. Any support given by the system would be of great help [RSC09]. For instance, the repository could cluster search results (see Section 4.2.4), then choose one representative per cluster (e.g. the medoid workflow), and finally let the user chose between those representatives. This approach can be applied recursively, allowing for an iterative query refinement. For instance, Stoyanovich et al. recently showed how data mining techniques can be used to cluster workflows based on the keywords appearing in the workflows description [STD10]. On the same line, users could be supported by selecting discriminative properties from similar workflows (like the particular tools used for the same task) and presenting them as binary choices to the user. Such choices could be augmented with techniques from collaborative filtering (users who have used this tool have then used this other tool [SK09]). Finally, the whole range of techniques developed for incorporating user feedback into a search should be considered [GF04].

4.3 Searching Workflow Runs

This subsection provides technical foundations to the needs presented in section 3.5 in which information about the executions of workflows (the runs) need to be exploited.

4.3.1 *Comparing Runs to find Points-of-Deviation*

Understanding why one workflow execution resulted in a different output than the execution of a similar or even identical workflow on the same data requires finding the point-in-time where the two runs have diverged. If the runs come from completely different workflows, the problem is similar to comparing two specifications (see Section 4.2). But when the two runs obey the same workflow specification then an algorithm may exploit the known common points to guide the placement of edit operations for aligning the runs. This is, or instance, exploited in the PDiffview system [ZCBD+09]. However, it will also occur that the runs come from different yet highly similar workflows (where one was created by substituting selected tasks performing the same logical operation), in which case a hybrid approach must be followed. This problem has, to our knowledge, not yet been tackled.

4.3.2 *Querying Workflow Provenance*

Recently, several approaches for querying workflow runs have emerged. Projects on querying provenance ([KSB10; MPB10]) are directly tailored to a specific workflow system (Kepler and Taverna, respectively). A major drawback of these approaches is that they do not consider similar traces. In the provenance community, other interesting approaches have been suggested [KIT10], but these are only considering data provenance. Whether or not these can be extended to work on workflow provenance graphs is an open issue.

4.4 Provenance for Workflow Specifications

Scientific experiments are by nature iterative, i.e., they often consist of various trials to refine the methodology. This fact also holds for scientific workflows, which often are refined, improved, extended etc. after their first usages. Knowing the evolution of a given workflow specification allows to understand the various hypotheses that the developer has tried to check. VisTrails [SVK+08] represents a workflow and its various trial-and-error steps in a tree in which each node is one state of a given workflow, i.e., they construct a phylogeny of workflow specifications. A survey on such topics can be found in [DR09].

4.5 Workflow Environment

Having chosen a workflow suitable for his needs, any true user would like to be able to run it as easily as possible. Having a workflow server attached to the repository taking in the data is possible but we believe that such an approach would fail in practice, as scientists may not be willing to give their data out of hand. Instead, it should be possible to download and run a chosen workflow locally. Such an approach raises several issues that have not yet been addressed properly.

First, the workflow system of choice must be installed locally. Second, all services to be performed in the workflow must be installed locally or accessible from the local machine. If tasks used web services, accessibility is less of an issue, but web services are inappropriate if large quantities of data need to be analyzed. Currently, in all other cases the local and tedious installation of further software is required. But ideally, a local workflow system running a given specification should automatically download and install all tools and data sets required. To this end, SciWFM need to be equipped with mechanisms for the dynamic downloading and installation of packages. This has not yet been addressed properly in the SciWFM community, though several solutions exist in other areas (see http://www.osgi.org/).

5. Conclusions

In this paper, we argued that future development in SciWFM should concentrate on the large class of user of analysis pipelines instead of the small class of people developing such pipelines. The core of our argument is that SciWFM in fact do not offer much advantage to developers when compared to scripting languages or programming environments such as R, especially when analysis pipelines get complicated (as it tends to be the case in bioinformatics). In contrast, an intuitive representation of existing analysis workflows combined with powerful methods of searching, adapting, and running them (a repository) would unravel the full potential of SciWFM – not to be used only as a design tool, but also and most importantly as a repository of solutions and, potentially, even as an environment of their executions. We described use cases that emerge from this viewpoint and derived a number of technical challenges that need to be addressed to develop comprehensive solutions.

Please note that, most of the additional functionality we have described can be very useful to power users too. Comparing and clustering workflows will allow highlighting two very important features of workflows: (i) their decomposition in main steps and (ii) their hierarchical structure. When a new workflow has to be designed, power users may thus save time by focusing only on the new step(s) they would have to add to an existing workflow instead of having to develop a complete pipeline from scratch. Another advantage for power users are the methods to study runs; comparing workflow runs may help power users better understand why an expected output has not been produced.

We are not the first to advocate such a movement; some isolated ideas and specific aspects have been described before [DRGS09; GDR+08]. However, we are the first to give the complete picture of the potential of a *scientific workflows repository* accompanied by a break-down of use cases into technical questions and a summary on algorithms, models and tools that already exist.

There are also a number of social issues that back-up our vision.

- We believe that the existence of such repositories would increase the pressure on scientific publishers to start demanding authors to submit their programs together with their findings – in the form of workflows [SBW+09]. Enforcement of co-submitting data with papers already is common-place in some areas (especially sequences and microarrays) and in active discussion for several other types of information [CCLS08; SG07]. Note that even depositing runs is not unrealistic, as these are the ultimate proof of what was done; for instance, the large DNA sequence repositories since long allow accompanying sequences with the so-called traces which are the primary output generated from a sequencing machine. Only these traces allow fully judging of the quality of the sequences.
- As briefly stated in the introduction, a comprehensive repository also is a strong incentive for developers to properly structure and describe their solutions. This increases their visibility and, combined with typical measures of such libraries as number of visits, number of derived workflows, number of associated results, user ratings etc., also raises reputation of the authors [DRGS09]. This, in turn, requires that workflows get citable, for instance by attaching DOI's to workflows in a repository.

In essence, we advocate to share and re-use workflows. Clearly, sharing programs is not a new idea. However, it is probably safe to say this idea does not work well when targeting arbitrary software artifacts, despite decades of research in software engineering on modularization, software repositories, component-based software etc. At the same time, there are many examples where it does work perfectly, especially when the domain of the programs to share is small. Prominent examples in the Life Sciences are the Open Bio* libraries which are used in many projects (biojava, biosql, bioperl, etc.). Another example is the statistical programming environment R – today, it is a de-facto standard that new analysis methods are published as R code and that they are integrated in programming libraries such as BioConductor [GCB+04] which in turn are used around the world. Similarly, a workflow repository like those we envision must be domain specific.

6. ACKNOWLEDGMENTS

We thank Bertram Ludäscher for many helpful comments on an earlier version of this, and Johannes Starlinger and Daniela Grigori for fruitful discussions on querying workflow repositories. This work was partly funded by the Deutsche Forschungsgemeinschaft (DFG), grant GRK-SOAMED.

7. REFERENCES

[AKD10] Ailamaki, A., Kantere, V. and Dash, D. (2010). "Managing scientific data." Communications of the ACM 53(6): 68-78.

[Alb09] Albrecht, A. (2009). "METL: Managing and Integrating ETL Processes". VLDB PhD workshop.

[ADMR05] Aumueller, D., Do, H., Massmann, S. and Rahm, E. (2005). "Schema and ontology matching with COMA++". SIGMOD Conference, Baltimore, US.

[AS10] Awad, A. and Sakr, S. (2010). "Querying Graph-Based Repositories of Business Process Models ". DASFAA workshops Tsukuba, Japan.

[ZCBD+09] Bao, Z., Cohen-Boulakia, S., Davidson, S. B., Eyal, A. and Khanna, S. (2009). "Differencing Provenance in Scientific Workflows". Int. Conf. on Data Engineering. Shanghai, China.

[BEKM08] Beeri, C., Eyal, A., Kamenkovich, S. and Milo, T. (2008). "Querying business processes with BP-QL." *Information Systems* **33**(6): 477-507.

[BTN+10] Bhagat, J., Tanoh, F., Nzuobontane, E., Laurent, T., Orlowski, J., Roos, M., Wolstencroft, K., Aleksejevs, S., Stevens, R., Pettifer, S.*, et al.* (2010). "BioCatalogue: a universal catalogue of web services for the life sciences." *Nucleic Acids Res* **38 Suppl**: W689-94.

[BCB+08] Biton, O., Cohen-Boulakia, S., Davidson, S. B. and Hara, C. S. (2008). "Querying and Managing Provenance through User Views in Scientific Workflows". Int. Conf. on Data Engineering. Cancún, México.

[CCLS08] Ceol, A., Chatr-Aryamontri, A., Licata, L. and Cesareni, G. (2008). "Linking entries in protein interaction database to structured text: The FEBS Letters experiment." *FEBS Letters* **582**(8): 1171-7.

[CBGZ06] Cesa-Bianchi, N., Gentile, C. and Zaniboni, L. (2006). "Incremental Algorithms for Hierarchical Classification." *Journal of Machine Learning Research* **7**: 31-54.

[CGH+06] Churches, D., Gombas, G., Harrison, A., Maassen, J., Robinson, C., Shields, M., Taylor, I. and Wang, I. (2006). "Programming scientific and distributed workflow with Triana services." *Concurrency and Computation: Practice and Experience* **18**(10): 1021-1037.

[BFL+04] Cohen-Boulakia, S., Froidevaux, C., Lair, S., Stransky, N., Radvanyi, F., Graziani, S. and Barillot, E. (2004). "Selecting Biomedical Data Sources According To User Preferences". Int. Conference on Intelligent Systems in Molecular Biology (ISMB/ECCB), Glasgow, UK.

[CBT09] Cohen-Boulakia, S. and Tan, W.-C. (2009). Provenance in Scientific Databases. In Liu, L. and Ozsu, M. T. (ed). Book "Encyclopedia of Database Systems", Springer, pp.

[CM90] Consens, M. P. and Mendelzon, A. O. (1990). "GraphLog: a Visual Formalism for Real Life Recursion". ACM Symposium on Principles of Database Systems, Nashville, Tennessee. pp 404-416.

[CH07] Cristianini, N. and Hahn, M. W. (2007). "Introduction to Computational Genomics - A Case Study Approach", Cambridge University Press.

[DR09] Dadam, P. and Rinderle, S. (2009). Workflow Evolution In Öszu, T. and Liu, L. (ed). Book "Encyclopedia of Database Systems ", Springer, pp.**:** 3540-3544.

[DF08] Davidson, S., B. and Freire, J. (2008). "Provenance and scientific workflows: challenges and opportunities". SIGMOD Conference. Vancouver, Canada.

[DRGS09] De Roure, D., Goble, C. and Stevens, R. (2009). "The design and realisation of the Virtual Research Environment for social sharing of workflows." *Future Generation Computer Systems* **25**(5): 561-567.

[DGST09] Deelman, E., Gannon, D., Shields, M. and Taylor, I. (2009). "Workflows and e-Science: An overview of workflow system features and capabilities." *Future Generation Computer Systems* **25**(5): 528-540.

[DSS+04] Deelman, E., Singh, G., Su, M.-H., Blythe, J., Gil, Y., Kesselman, C., Mehta, G., Vahi, K., Berriman, G. B., Good, J*., et al.* (2004). "Pegasus: A framework for mapping complex scientific workflows onto distributed systems." *Scientific Programming* **13**(3): 219-237.

[DKM+05] Delcambre, L., Kop, C., Mayr, H., Mylopoulos, J., Pastor, O., Bowers, S. and Ludäscher, B. (2005). Actor-Oriented Design of Scientific Workflows. In (ed). Book "Conceptual Modeling – ER 2005", Springer Berlin / Heidelberg, pp.**:** 369-384.

[GCB+04] Gentleman, R. C., Carey, V. J., Bates, D. M., Bolstad, B., Dettling, M., Dudoit, S., Ellis, B., Gautier, L., Ge, Y., Gentry, J.*, et al.* (2004). "Bioconductor: open software development for computational biology and bioinformatics." *Genome Biol* **5**(10): R80.

[GGW+09] Gibson, A., Gamble, M., Wolstencroft, K., Oinn, T., Goble, C., Belhajjame, K. and Missier, P. (2009). "The data playground: An intuitive workflow specification environment." *Future Generation Computer Systems* **25**(4): 453-459.

[GS08] Goble, C. and Stevens, R. (2008). "State of the nation in data integration for bioinformatics." *J Biomed Inform* **41**(5): 687-93.

[GDR+08] Goderis, A., De Roure, D., Goble, C., Bhagat, J., Cruickshank, D., Fisher, P., Michaelides, D. and Tanoh, F. (2008). "Discovering scientific workflows: the myExperiment benchmarks." *IEEE Transactions on Automation Science and Engineering*.

[GNT+10] Goecks, J., Nekrutenko, A. and Taylor, J. (2010). "Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences." *Genome Biol* **11**(8): R86.

[GCB08] Grigori, D., Corrales, J. C. and Bouzeghoub, M. (2008). "Behavioral matchmaking for service retrieval: application to conversation protocols." *Information Systems* **33**(7-8): 681-698.

[GF04] Grossmann, D. A. and Frieder, O. (2004). "Information Retrieval - Algorithms and Heuristics", Springer.

[Gus97] Gusfield, D. (1997). "Algorithms on Strings, Trees and Sequences", Cambridge University Press.

[HS08] He, H. and Singh, A. K. (2008). "Graphs-at-a-time: query language and access methods for graph databases". SIGMOD Conference, Vancouver, Canada pp 405-418.

[HSRC08] Ho, J. W., Stefani, M., dos Remedios, C. G. and Charleston, M. A. (2008). "Differential variability analysis of gene expression and its application to human diseases." *Bioinformatics* **24**(13): i390-8.

[HMB07] Howe, B., Maier, D. and Bright, L. (2007). "Smoothing the ROI curve for scientific data management applications". Conf. on Innovative Data Research, Asilomar, USA.

[KIT10] Karvounarakis, G., Ives, Z. G. and Tannen, V. (2010). "Querying data provenance". SIGMOD Conference, Indianapolis, US.

[KSB10] Kumar, A. M., Bowers, S. and Ludaescher, B. (2010). "Techniques for efficiently querying scientific workflow provenance graphs". Int. Conf. on Extending Database Technology. Lausanne, Switzerland.

[LSC10] Liu, Z., Shao, Q. and Chen, Y. (2010). "Searching Workflows with Hierarchical Views." *PVLDB* **3**(1): 918-927.

[LKMS08] Lottaz, C., Kostka, D., Markowetz, F. and Spang, R. (2008). "Computational diagnostics with gene expression profiles." *Methods Mol Biol* **453**: 281-96.

[LAB+05] Ludaescher, B., Altintas, I., Berkley, C., Higgins, D., Jaeger, E., Jones, M., Lee, E. A., Tao, J. and Zhao, Y. (2005). "Scientific workflow management and the Kepler system." *Concurrency and Computation: Practice and Experience* **18**(10): 1039-1065.

[LAG03] Ludaescher, B., Altintas, I. and Gupta, A. (2003). "Compiling Abstract Scientific Workflows into Web Service Workflows". 15th Int. Conf. on Scientific and Statistical Database Management, Cambridge, US.

[MLB+10] Missier, P., Ludascher, B., Bowers, S., Dey, S., Sarkar, A., Shrestha, B., Altintas, I., Anand, M. K. and Goble, C. (2010).

"Linking multiple workflow provenance traces for interoperable collaborative science". 5th Workshop on Workflows in Support of Large-Scale Science, New Orleans, US.

[MPB10] Missier, P., Paton, N. W. and Belhajjame, K. (2010). "Fine-grained and efficient lineage querying of collection-based workflow provenance". 13th Int. Conf. on Extending Database Technology. Lausanne, CH.

[MLA+08] Moreau, L., Ludaescher, B., Altintas, I., Barga, R. S., Bowers, S., Callahan, S., Chin, G. J. R., Clifford, B., Cohen, S., Cohen-Boulakia, S.*, et al.* (2008). "Special Issue: The First Provenance Challenge." *Concurrency and Computation: Practice and Experience* **20**(5): 409-418.

[OGA+05] Oinn, T., Greenwood, M., Addis, M., Alpdemir, M. N., Ferris, J., Glover, K., Goble, C., GODERIS, A., HULL, D., MARVIN, D.*, et al.* (2005). "Taverna: Lessons in creating a workflow environment for the life sciences." *Concurrency and Computation: Practice and Experience* **18**(10): 1067-1100.

[RLS+06] Radetzki, U., Leser, U., Schulze-Rauschenbach, S. C., Zimmermann, J., Lussem, J., Bode, T. and Cremers, A. B. (2006). "Adapters, shims, and glue - service interoperability for in silico experiments." *Bioinformatics* **22**(9): 1137-43.

[RFL06] Reisig, W., Fahland, D., Lohmann, N., Massuthe, P., Stahl, C., Weinberg, D., Wolf, K. and Kaschner, K. (2006). "Analysis Techniques for Service Models". 2nd Int. Symp. on Leveraging Applications of Formal Methods, Verification and Validation. pp 11-17.

[RSC09] Ren, K., Sarvas, R. and Calic, J. (2009). "Interactive search and browsing interface for large-scale visual repositories." *Multimedia Tools and Applications* **49**(3): 513-528.

[SVK+08] Scheidegger, C., E. , Vo, H., T. , Koop, D., Freire, J. and Silva, C., T. (2008). "Querying and re-using workflows with VisTrails". SIGMOD Vancouver, Canada.

[SBW+09] Schofield, P. N., Bubela, T., Weaver, T., Portilla, L., Brown, S. D., Hancock, J. M., Einhorn, D., Tocchini-Valentini, G., Hrabe de Angelis, M. and Rosenthal, N. (2009). "Post-publication sharing of data and tools." *Nature* **461**(7261): 171-3.

[SMH+10] Seligman, L., Mork, P., Halevy, A., Smith, K., Carey, M. J., Chen, K., Wolf, C., Madhavan, J. and Kannan, A. (2010). "OpenII: An Open Source Information Integration Toolkit". Int. Conf. on Very Large Databases, Singapore.

[SG07] Seringhaus, M. R. and Gerstein, M. B. (2007). "Publishing perishing? Towards tomorrow's information architecture." *BMC Bioinformatics* **8**: 17.

[Ste08] Stein, L. D. (2008). "Towards a cyberinfrastructure for the biological sciences: progress, visions and challenges." *Nature Reviews Genetics* **9**(9): 678-88.

[STD10] Stoyanovich, J., Taskar, B. and Davidson, S. (2010). "Exploring Repositories of Scientific Workflows". Int. Workshop on Workflow Approaches to New Data-centric Science Indianapolos, US.

[SK09] Su, X. and Khoshgoftaar, T. M. (2009). "A Survey of Collaborative Filtering Techniques." *Advances in Artificial Intelligence*.

[WYDM04] Wu, W., Yu, C., Doan, A. and Meng, W. (2004). "An interactive clustering-based approach to integrating source query interfaces on the deep Web". SIGMOD Conference, Paris, France.

[ZL10] Zimmermann, K. and Leser, U. (2010). "Analysis of Affymetrix Exon Arrays". Technical Report 235, Department for Computer Science, Humboldt-Universität zu Berlin.

A Survey on Energy-Efficient Data Management

Jun Wang, Ling Feng
Dept. of CS&T, Tsinghua Univ., Beijing, China
wjun09@mails.tsinghua.edu.cn
fengling@tsinghua.edu.cn

Wenwei Xue, Zhanjiang Song
Nokia Research Center, Beijing, China
wayne.xue@nokia.com
zhanjiang.song@nokia.com

ABSTRACT

Energy management has now become a critical and urgent issue in green computing. A lot of efforts have been made on energy-efficiency computing at various levels from individual hardware components, system software, to applications. In this paper, we describe the energy-efficiency computing problem, as well as possible strategies to tackle the problem. We survey some recently developed energy-saving data management techniques. Benchmarks and power models are described in the end for the evaluation of energy-efficiency solutions.

1. INTRODUCTION

Now and in the future, green computing will be a key challenge for both information technology and business. Green computing aims at environmentally sustainable computing and responsible use of computers and related resources. Murugesan *et al.* defined the field of green computing as "*the study and practice of designing, manufacturing, using, and disposing of computers, servers, and associated subsystems such as monitors, printers, storage devices, and networking and communications systems efficiently and effectively with minimal or no impact on the environment* [28]."

With the limited primary sources of energy and rapid climbing of energy demanded by computing, the commitment to reduce power consumption and environmental impact becomes increasingly important. *Energy efficiency* thus constitutes a focal point for green computing.

In fact, most people in the world today are aware of the energy problem at a high level: even if our primary sources of energy are running out, the demand for energy in both commercial and domestic environments is increasing, and the side effect of consistent energy use influences negatively our global environment. Based on the report of the US Environmental Protection Agency, "*the servers and data centers in USA alone consumed about 61 billion kilowatt-hours (kWh) at a cost of $4.5 billion, which was about 1.5% of the total U.S. electricity consumption in 2006, and this energy consumption is expected to double by 2011 if continuously powering computer servers and data centers using the same methods* [11]." Xu *et al.* showed that electricity consumed by computer servers and cooling systems in a typical data center contributes to around 20 percent of the total ownership cost, equivalent to one-third of the total maintenance cost [42]. When a data center reaches its maximum provisioned power, it has to be replaced or augmented at a great expense [33]. In the very near future, energy efficiency is expected to be one of the key purchasing arguments in the society.

Nowadays, power and energy have started to severely constrain the design of components, systems, computing clusters, data centers, and applications. Better equipment design and better energy management policies are desirable to address these concerns.

More and more computer designers and users are concerned about energy-efficient computing. In this survey article, we overview these great efforts, with an emphasis on energy-efficient data management, which was ignored and starts to get attention very recently.

The remainder of this paper is organized as follows. In Section 2, the energy-efficiency computing problem and some guidelines to tackle the problem are discussed. In Section 3, we overview work done on energy-efficient data management. Evaluation techniques on energy performance including benchmarks and power models are described in Section 4. We conclude the paper in section 5.

2. FUNDAMENTALS OF ENERGY EFFICIENT COMPUTING

In this section, after a brief description of the energy-efficiency problem, we outline some feasible ways to tackle the problem.

2.1 Energy-Efficiency Problem

Energy consumption can be generally defined as:

$$Energy = AvgPower \times Time$$

where $Energy$ and $AvgPower$ are measured in $Joule$ and $Watt$, respectively, and $1\ Joule = 1\ Watt \times 1\ Second$.

Energy efficiency is equivalent to the ratio of performance, measured as the rate of work done, to the power used [37] and the performance can be represented by response time or throughput of the computing system.

$$\begin{aligned} Energy\ Efficiency &= \frac{Workdone}{Energy} = \frac{Workdone}{Power \times Time} \\ &= \frac{Performance}{Power} \end{aligned} \tag{1}$$

The main approach towards energy-efficiency is efficient power management. According to equation (1), there are two ways to enhance energy-efficient computing: either improving the performance with the same power, or reducing power consumption without sacrificing too much performance. For energy-efficient systems, while maximal performance for some tasks (or the whole workload) is still desirable in some cases, the systems must also ensure the energy usage is minimized. Preferably, a computing system consumes the minimum amount of energy to perform a task at the maximal performance level [10].

Note that the relationship between performance and energy efficiency is not mutually exclusive. A maximal performance could also be achieved by deactivating some resources or lowering certain individual performance without affecting the workload's best possible completion time or throughput in order to optimize energy usage. Brown *et al.* treated energy efficiency as an optimization problem [10]. To minimize the total energy, an energy-efficient system must adjust the system's hardware resources dynamically, so that only what is needed to execute tasks is made available. Rivoire *et al.* pointed out two major complementary ways to solve the energy-efficiency problem: either building energy efficiency into the initial design of computer components and systems, or adaptively managing the power consumption of systems or groups of systems in response to changing conditions related to the workload or environment [36].

2.2 Solution Guidelines

To deliver effective solutions to the energy-efficiency problem, the following six considerations can be taken as the solution design guidelines.

1) *Comprehensive Examination of System Components.* To save power consumption, we shall first investigate where the power is spent and how to optimize the power usage. Within a computer system, there are generally five energy consumers, namely, *processor, disk, memory, I/O devices,* and *chipset.* Achieving energy-efficiency requires improvements in the energy usage profile of every system component.

2) *Adopting Power-Manageable Hardware Components.* Adopting power-manageable hardware components could help improve energy-efficiency. For example, the voltage of hardware components can be increased or decreased through dynamic voltage scaling (DVS), which is a power management technique in computer architecture, depending upon circumstances. Dynamic voltage scaling to decrease voltage is known as undervolting, and this situation can conserve power [12]. In addition, employing small form factor disk drives, solid state disk drives, large memory configurations, low power processors and memories could decrease power consumption [31]. HP and IDC also estimated that about 69 percent energy reduction can be achieved within a three-year period for IT organizations that migrate to blade self-contained architecture, where blades can span from servers and storage devices to workstations and virtual desktops [20, 1].

3) *Building Power Models for Computing Systems.* Also, one needs to know how a computing system is constructed and how an energy-efficient system operates. It is important to construct a power model that allows the system to know how the power is consumed, and how the system can manipulate and tune that power [10].

4) *Understanding and Measuring System Performance.* To counter for performance with the least power consumption, computing systems must have ways to timely understand and measure system performance related to task execution under different dynamic workloads.

5) *Constructing Energy Optimizers.* The system must accommodate an energy optimizer component, which is responsible for an energy-efficient hardware configuration throughout the system operation at all times. The optimization approaches may be based on either heuristic or analytical techniques, as indicated by Brown *et al.* in [10].

6) *Reducing Peak Power.* Barroso *et al.* explained that current desktop and server processors can consume less than one-third of their peak power at very low activity modes, which can thus save around 70 percent of peak power [7]. Tsirogiannis *et al.* indicated that almost 50 percent of peak power is actually consumed at idle [37].

3. ENERGY-EFFICIENT DATA MANAGEMENT

Over the past few decades, performance remains as the main goal of database management systems (DBMSs). In light of an increasing concern about energy, energy management starts to draw attention of the database community. The 2008 Claremont report on database research emphasized explicitly the importance of *power-aware DBMSs that limit energy costs without sacrificing scalability* [4]. To achieve energy-efficient DBMSs, solutions from both hardware and software perspectives are desirable. That is, DBMSs not only need to accurately estimate and online measure the hardware energy consumption characteristics under both static and dynamic loads, but also dynamically adapt and tune data management strategies to meet response time and energy goals when initial prediction for energy consumption deviates from the real case [25, 18, 42, 41, 37].

In this section, we describe efforts on energy-efficient data management at server, sensor, and mobile sides.

3.1 Server Side

Research on energy-efficient data management at server side so far mainly considers the most fundamental database operation - query processing and optimization.

3.1.1 Hardware-Based Approach

Under the assumptions that a CPU consumes a significant amount of power compared to other components in a database system, and the performance of speed and energy of hard disk drive is close but does not exactly follow the change in the read block size, Lang *et al.* proposed a PVC (Processor Voltage/Frequency Control) mechanism to trade energy consumption for performance [25]. It aims to execute instructions at a lower processor voltage and frequency by leveraging the ability of modern processors. The CPU frequency is determined by two settings: the front side bus (FSB) speed and the CPU multiplier. There are basically two methods to modulate CPU frequency, namely, p-state transitioning and underclocking. P-states are characterized by the combination of CPU multiplier and CPU voltage settings. Underclocking has the ability to more finely tune the CPU speed by slowing the FSB speed. [25] adopted underclocking to modulate the CPU frequency, and their experiments showed that PVC can be used to reduce the CPU energy consumption by 20% and 49%, while incurring 6% and 3% response time penalties on MySQL and a commercial DBMS, respectively.

3.1.2 Software-Based Approaches

Hardware-based approaches constitute only part of solutions. Considering hardware heterogeneity and limited power knobs that most hardware offers today, data management software shall play an effective role in energy optimization as well. Physical data independence and query optimization of DBMSs do provide opportunities for software-level control over power-performance tradeoffs [18].

Harizopoulos *et al.* categorized three kinds of software-based approaches for reducing energy in DBMSs [18].

(1) *Energy-aware optimization*, i.e., using existing system-wide knobs and internal query optimization parameters to achieve the most energy-efficient configuration for the underlying hardware. Xu *et al.* gave a strategy to find query plans with low power costs [42, 41]. To do that, a static power profile for each basic database operation in query processing is defined and maintained as system parameters of DBMSs. The power cost can be obtained from the specifications of hardware components and divided by related estimated time through an iterative approach. The power cost of a plan can be calculated from those of the higher-level operations, containing such basic operations like CPU power cost per tuple/indexed tuple, power cost for reading/writing one page without buffering, and so on. Different power cost functions can thus be constructed for accessing single relation via different access methods and join operations.

(2) *Resource use consolidation*, i.e., shifting computation and relocating data to consolidate resource use in time and space. Whenever system resources are not fully utilized, the system may allow other concurrent tasks to utilize the resources or allow the resource to enter a suspended or reduced power mode to save energy.

(3) *Redesign software components* to minimize energy use, reduce code bloat, and sacrifice certain properties (or allow under-perform in certain metrics) to improve energy efficiency. For instance, Lang *et al.* proposed a QED (Improved Query Energy-efficiency by Introducing Explicit Delays) mechanism, which uses query aggregation to leverage common components of queries in a workload. In QED, queries are delayed and placed into a queue on arrival. When the queue reaches a certain threshold, all the queries in the queue are examined to determine if they can be aggregated into a small number of groups, such that queries in each group can be evaluated together [25]. On a workload with sim-

ple selection queries on MySQL, QED saves 54% of the CPU energy consumption while increasing the average query response time by 43%.

3.1.3 Tradeoff Between Energy and Performance

For getting the greatest energy efficiency in DBMSs, we must find the relationship between energy (or power) and performance. It is interesting to note two different opinions in recent research work.

- [42, 25, 18] claimed that energy efficiency and performance are two different optimization goals, and there exists the tradeoff of energy efficiency and performance.
- [37] claimed that energy efficiency and performance are consistent, and it said that "*within a single node system (intended for use in scale-out architectures), the most energy-efficient configuration is typically the highest performing one.*"

The cause of appearing the above two different statements may be their different assumptions and different estimation methods used. In the first line of work, the baseline power (i.e., idle power) is not included in calculating the energy cost of the DBMS after processing the workload. The second work considered CPU power only rather than the system's overall active power. By experiments, they found out that the CPU power does not vary linearly with CPU utilization (which is also the number of cores in the multi-core machine), and utilization is a poor proxy for CPU power. The CPU power used by various operators can vary up to 60%, even when they have the same utilization. By measuring the power of system components from idle to full utilization, almost 50% of peak power is consumed at idle regardless of query complexity and strategy. This fixed power cost adds a large constant term to the denominator in energy-efficiency equation (1), which makes all subsequent relative power increases worth the added performance, especially when the dynamic power range is small [37]. However, in DBMSs, there do exist queries that require many CPU operations, as well as I/O-intensive queries. Considering CPU alone, the margin of improvements might be greatly reduced by factoring in the power costs of all components.

3.2 Sensor Side

Nowadays, wide applications of sensors in health care, agri-food, environmental and security sectors call for effective sensor data processing and management techniques. As the power source of sensors often comes from a battery with a limited energy, reducing energy consumption of sensors is critical to provide a long enough lifetime service and avoid inconvenient replacement of the battery in a hostile or unpractical environment. For a sensor node, the communication cost is often several orders of magnitude higher than the computation cost, and in wireless sensor networks, the majority of the energy is actually consumed for sensor's communication rather than computation. Reducing communication workloads among sensor nodes is apparently the most effective energy-efficient operation.

Anastasi *et al.* have made a very good comprehensive survey on existing energy-efficient sensing techniques, including *duty cycling*, *data-driven*, and *mobility-based schemes*. Detailed description of each scheme can be found in [5].

1) *Duty cycling schemes* are oblivious to data sampled by sensor nodes. Radio transceiver nodes are put in a sleep mode whenever communication is not required. Duty cycling schemes can be achieved through two different and complementary approaches, i.e., *topology control* and *power management*. The topology control mechanism dynamically adapts the network topology based on the application needs so as to allow network operations while minimizing the number of active nodes. The power management mechanism aims at certain sensor nodes, which can sleep or wake-up by an *on-demand, scheduled rendezvous,* or *asynchronous* protocol [5].

2) *Data-driven schemes* are designed to reduce the amount of sampled data without sacrificing sensing accuracy required by applications through *data reduction* and *energy-efficient data acquisition. Data reduction* approaches address the case of unneeded samples. Techniques developed involve in-network processing, data compression, and data prediction. *Energy-efficient data acquisition* approaches mainly focus on reducing the energy spent in sensing by means of adaptive, hierarchical, and model-based active sampling [5]. For instance, to reduce the communication cost, Jain *et al.* proposed a Dual Kalman Filter architecture as a general and adaptive filtering solution to the stream resource management problem [21]. The architecture lays two equal Kalman Filters between clients and servers. The dual filters predict future data values. Only when the filter at the remote source fails to predict future data within the precision constraint, the sensor sends update value to the server.

3) Mobility has been considered as an alternative solution for energy efficient data collection in wireless sensor networks. The sensors are equipped with mobilities for changing their location. *Mobility-based schemes* can be classified as mobile-sink and mobile-

relay schemes, depending on the type of the mobile entity [5].

3.3 Mobile Side

Advancement in computing and communication has led to an increased use of a large number of mobile devices. Mobile devices are equipped with more computing and sensing facilities include GPS, WiFi, Bluetooth, accelerometers, audio, video, light sensors, and so on. These embedded sensors in mobile devices are also major power consumers, while the devices have still limited power sources such as batteries. Recently, the problem of energy-efficient data management on mobile devices has received much attention. Whang *et al.* proposed to support functionalities with low power consumption, which is also a crucial requirement for a ubiquitous database management systems [39].

Energy-efficient techniques developed at the mobile side center around *position tracking, continuous context monitoring,* and *complex event detection.*

1) *Position tracking* is an important feature of a modern mobile device. The most common method is to use GPS. However, GPS is extremely power hungry. To minimize energy consumption and improve accurate, Kjærgaard *et al.* and Paek *et al.* proposed to periodically duty-cycle GPS [24, 30]. The idea is to use historic information to estimate and predict users' movement while concurrently utilizing other sensors such as accelerometer and Bluetooth. In a mobile broadcast environment, index has extensively been adopted to support efficient location-based data access and query [43, 13, 40], since efficient indexing structures can contribute significantly to reduce the tuning time, which is frequently used to estimate the power consumption of a mobile client.

2) For *continuous context monitoring*, Kang *et al.* proposed a middle-tier framework between applications and embedded sensors in a mobile environment [23]. Wang *et al.* also presented an energy efficient mobile sensing framework to automatically recognize user state and detect state transition [38]. These two works also turned on a minimum set of sensors and used appropriate sensor duty cycles to reduce unnecessary, expensive computation and communication in the context monitoring process for energy savings.

3) With the widespread use of wireless connectivity and end-user mobile devices, *complex event detection* can be carried out by mobile devices. Considering the interaction between front-end and server, Neophytou *et al.* proposed three power-aware query operator placement algorithms that determine which part of a continuous query plan is executed at the stream management server and which part is executed at the users' wireless devices [29]. Gredik *et al.* also presented a distributed real-time approach to monitor moving object. They utilized the computational power at mobile objects to alleviate server-side load and communication cost [16].

4. EVALUATION METHODS

How to compare different energy-efficient computing methods against each other and how to estimate the developed methods whether or not they correspond to reality constitutes another important question. Substantial efforts on evaluation metrics and models for energy-efficiency computing have also been made in both academia and industry.

4.1 Models and Metrics

The real power consumption of a specific system depends on many factors, such as workload, system balance, and environmental parameters. Measuring power consumption needs accurate power and thermal models on individual components, systems, data and computing centers, and applications.

Three different types of modeling approaches exist in the literature, including *simulation-based*, *detailed analytical*, and *high-level black-box* approaches [33, 34].

1) Considering the difficulty in obtaining detailed knowledge about many components in a full system, *simulation-based* approaches intend to model individual components rather than the whole system or a collection of systems by simulation [17, 19, 9].

2) Without simulation, *detailed analytical* approaches periodically collect hardware and software metrics [6, 22]. For example, Bircher *et al.* used microprocessor performance counters for online measurement of complete system power consumption [8]. Xu *et al.* presented a power model to accurately measure the energy costs of database query execution plans [42, 41] with the hypothesis that *the peak power consumption of an entire system during the measurement interval is identical to the aggregate of the individual nameplate power consumption.*

3) *High-level black-box* approaches construct a real-time model by fitting a model to the real-time metrics collected without relying on implementation knowledge. For example, Economou *et al.* used a one-time calibration phase to generate a power consumption model by correlating AC power measurements with user-level system utilization metrics at a system level [14]. Fan *et al.* aggregated power usage of large collections of servers for different classes of applications over history data [15]. Meisner *et al.* incorporated suspending and waking transitions

to the power model [27]. Lang *et al.* proposed a mathematical model for the energy consumption of a MapReduce cluster, which adopted the workload characteristics and hardware characteristics as abstract meta-models [26]. Poess *et al.* developed a power consumption model based on data readily available in the TPC-C full disclosure report of published benchmarks [31].

4.2 Benchmarks

Researchers, governmental agencies, and industry standard consortia for performance measurements, including Transaction Processing Performance Council (TPC), Standard Performance Evaluation Corporation (SPEC), and Storage Performance Council (SPC), have also actively developed benchmarks to measure energy consumption of computer systems. Poess *et al.* provided a very comprehensive overview of the currently available energy benchmarks, and analyzed their commonalities and differences along various dimensions, including hardware components, workload and type of application, along with metric attributes and accuracy and calibration requirements [31, 32].

Basically, there are two types of energy benchmarks being developed so far. They are *specialized benchmarks* and *extended benchmarks* with additional energy metrics added to the existing benchmarks.

For example, TPC established a working group which adds energy efficiency metrics to all its benchmarks. The primary metric, reported by TPC-Energy [3], is in the form of "*Watts per Performance*" for the overall System Under Test (SUT), where the performance unit is specific to each TPC Benchmark. SPEC presented the SPECpower_ssj2008 [2] benchmark, which examined the relationship of power and performance and power consumption for servers at different performance levels, spanning from 100 percent utilization to idle in 10 percent segments, over a set period of time [32].

Rivoire *et al.* developed a benchmark called *JouleSort* for evaluating energy efficiency of various sorting algorithms. It used the same workload as the other external sort benchmarks, but its metric covered total energy, which is a combination of power consumption and performance. *JouleSort* is now an I/O-centric benchmark that measures the energy efficiency of systems at peak use [35].

5. CONCLUSION

In this survey, we present the concept and challenges of energy-efficiency problem in green computing. We focus on energy-efficient data management at server, sensor, and mobile sides in this paper. Various power models and energy benchmarks are also presented in the paper.

Acknowledgment

The work is funded by National Natural Science Foundation of China (60773156, 61073004) and Chinese Major State Basic Research Development 973 Program (2011CB302203-2).

6. REFERENCES

[1] HP Blade System c-Class portfolio. *http://hl8004.wwwl.hp.com/products/blades/components/c-class-components.html.*

[2] SPECpower_ssj2008. *http://www.spec.org/power_ssj2008/.*

[3] TPC Energy Specification Version 1.2.0. *http://www.tpc.org/tpc_energy/spec/.*

[4] R. Agrawal, A. Ailamaki, P. Bernstein, E. Brewer, M. Carey, S. Chaudhuri, A. Doan, D. Florescu, M. Franklin, H. Garcia-Molina, et al. The claremont report on database research. *Communications of the ACM*, 52(6):56–65, 2009.

[5] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella. Energy conservation in wireless sensor networks: A survey. *Ad Hoc Networks*, 7(3):537–568, 2009.

[6] R. Azimi, M. Stumm, and R. Wisniewski. Online performance analysis by statistical sampling of microprocessor performance counters. In *ICS*, pages 101–110. ACM, 2005.

[7] L. Barroso and U. Holzle. The case for energy-proportional computing. *IEEE Computer*, 40(12):33, 2007.

[8] W. Bircher and L. John. Complete system power estimation: A trickle-down approach based on performance events. In *ISPASS*, pages 158–168, 2007.

[9] D. Brooks, V. Tiwari, and M. Martonosi. Wattch: a framework for architectural-level power analysis and optimizations. *ACM SIGARCH Computer Architecture News*, 28(2):94, 2000.

[10] D. Brown and C. Reams. Toward energy-efficient computing. *Communications of the ACM*, 53(3):50–58, 2010.

[11] R. Brown, E. Masanet, B. Nordman, B. Tschudi, A. Shehabi, J. Stanley, J. Koomey, D. Sartor, P. Chan, J. Loper, et al. Report to congress on server and data center energy efficiency. *Public law*, pages 109–431, 2007.

[12] J. Chen and C. Kuo. Energy-efficient scheduling for real-time systems on dynamic voltage scaling (DVS) platforms. In *RTCSA*, pages 28–38. IEEE, 2007.

[13] Y. Chung, S. Yoo, and M. Kim. Energy and Latency Efficient Processing of Full-text Searches on a Wireless Broadcast Stream. *TKDE*, 22(2):207, 2010.

[14] D. Economou, S. Rivoire, C. Kozyrakis, and P. Ranganathan. Full-system power analysis and modeling for server environments. In *MoBS*. Citeseer, 2006.

[15] X. Fan, W. Weber, and L. Barroso. Power provisioning for a warehouse-sized computer. In *ISCA*, page 23. ACM, 2007.

[16] B. Gedik and L. Liu. Mobieyes: Distributed processing of continuously moving queries on moving objects in a mobile system. In *EDBT*, pages 523–524. Springer, 2004.

[17] S. Gurumurthi, A. Sivasubramaniam, M. Irwin, N. Vijaykrishnan, M. Kandemir, T. Li, and L. John. Using complete machine simulation for software power estimation: The softwatt approach. In *HPCA*. IEEE, 2002.

[18] S. Harizopoulos, M. Shah, J. Meza, and P. Ranganathan. Energy efficiency: The new holy grail of data management systems research. In *CIDR*, 2009.

[19] T. Heath, A. Centeno, P. George, L. Ramos, Y. Jaluria, and R. Bianchini. Mercury and freon: temperature emulation and management for server systems. *ACM SIGARCH Computer Architecture News*, 34(5):116, 2006.

[20] IDC White Paper. Forecasting Total Cost of Ownership for Initial Deployments of Server Blades. *ftp://hp.pl/pub/c-products/blades/idc-tco-deployment.pdf*, 2006.

[21] A. Jain, E. Chang, and Y. Wang. Adaptive stream resource management using kalman filters. In *SIGMOD*, pages 11–22. ACM, 2004.

[22] R. Joseph and M. Martonosi. Run-time power estimation in high performance microprocessors. In *ISLPED*, pages 135–140. ACM, 2001.

[23] S. Kang, J. Lee, H. Jang, H. Lee, Y. Lee, S. Park, T. Park, and J. Song. Seemon: scalable and energy-efficient context monitoring framework for sensor-rich mobile environments. In *MobiSys*, pages 267–280. ACM, 2008.

[24] M. Kjærgaard, J. Langdal, T. Godsk, and T. Toftkjær. Entracked: energy-efficient robust position tracking for mobile devices. In *MobiSys*, pages 221–234. ACM, 2009.

[25] W. Lang and J. Patel. Towards eco-friendly database management systems. In *CIDR*, 2009.

[26] W. Lang and J. Patel. Energy Management for MapReduce Clusters. *Proceedings of the VLDB Endowment*, 3(1), 2010.

[27] D. Meisner, B. Gold, and T. Wenisch. PowerNap: eliminating server idle power. *ACM SIGPLAN Notices*, 44(3):205–216, 2009.

[28] S. Murugesan. Harnessing green it: Principles and practices. *IT professional*, 10(1):24–33, 2008.

[29] P. Neophytou, M. Sharaf, P. Chrysanthis, and A. Labrinidis. Power-Aware Operator Placement and Broadcasting of Continuous Query Results. In *MobiDE*. ACM, 2010.

[30] J. Paek, J. Kim, and R. Govindan. Energy-Efficient Rate-Adaptive GPS-based Positioning for Smartphones. In *MobiSys*, 2010.

[31] M. Poess and R. Nambiar. Energy cost, the key challenge of today's data centers: a power consumption analysis of TPC-C results. *Proceedings of the VLDB Endowment*, 1(2):1229–1240, 2008.

[32] M. Poess, R. Nambiar, K. Vaid, J. Stephens Jr, K. Huppler, and E. Haines. Energy benchmarks: a detailed analysis. In *the 1st International Conference on Energy-Efficient Computing and Networking*, pages 131–140. ACM, 2010.

[33] P. Ranganathan, S. Rivoire, and J. Moore. Models and metrics for energy-efficient computing. *Advances in Computers*, 75:159–233, 2009.

[34] S. Rivoire, P. Ranganathan, and C. Kozyrakis. A comparison of high-level full-system power models. *HotPower. USENIX Association*, 2008.

[35] S. Rivoire, M. Shah, P. Ranganathan, and C. Kozyrakis. JouleSort: a balanced energy-efficiency benchmark. In *SIGMOD*, page 376. ACM, 2007.

[36] S. Rivoire, M. Shah, P. Ranganathan, C. Kozyrakis, and J. Meza. Models and metrics to enable energy-efficiency optimizations. *IEEE Computer*, 40(12):39, 2007.

[37] D. Tsirogiannis, S. Harizopoulos, and M. Shah. Analyzing the energy efficiency of a database server. In *SIGMOD*, pages 231–242. ACM, 2010.

[38] Y. Wang, J. Lin, M. Annavaram, Q. Jacobson, J. Hong, B. Krishnamachari, and N. Sadeh. A framework of energy efficient mobile sensing for automatic user state recognition. In *MobiSys*, pages 179–192. ACM, 2009.

[39] K. Whang, I. Song, T. Kim, and K. Lee. The ubiquitous DBMS. *ACM SIGMOD Record*, 38(4):14–22, 2009.

[40] J. Xu, B. Zheng, W. Lee, and D. Lee. Energy efficient index for querying location-dependent data in mobile broadcast environments. In *ICDE*, pages 239–250, 2003.

[41] Z. Xu. Building a power-aware database management system. In *IDAR*, pages 1–6. ACM, 2010.

[42] Z. Xu, Y. Tu, and X. Wang. Exploring Power-Performance Tradeoffs in Database Systems. In *ICDE*, pages 485–496. IEEE, 2010.

[43] B. Zheng, W. Lee, K. Lee, D. Lee, and M. Shao. A distributed spatial index for error-prone wireless data broadcast. *VLDB Journal*, 18(4):986, 2009.

Laura Haas Speaks Out

on managing teams versus children, research versus product development, and much more

by Marianne Winslett and Vanessa Braganholo

Laura Haas
https://researcher.ibm.com/researcher/view.php?person=almaden-laura

Welcome to this installment of ACM SIGMOD Record's series of interviews with distinguished members of the database community. I'm Marianne Winslett, and today we are in Indianapolis, site of the 2010 SIGMOD and PODs conference. I have here with me Laura Haas, who is the Director of Computer Science at IBM Almaden Research Center. Laura is an ACM Fellow, an IBM Fellow, a recipient of the SIGMOD Contributions Award, and a member of the National Academy of Engineering. Her PhD is from the University of Texas at Austin. So, Laura, welcome!

In the US, NAE membership is the engineering field's greatest honor. Is it true that you did the work that led to your induction because you wanted to get out of creating a version of R for PCs?*

Yes, in some sense it is! We were asked to build a version of the relational database that would run on a PC, back in the days when PCs had 640K of memory, and very limited space, and all these tight constraints. It was all architectural, and understanding the hardware, and the bits and the bytes, and I've never been particularly interested in the low level details like that. I was always interested in the abstraction layer. So we were kind of looking around for what else we could do while we were doing this. Then we decided we were going to have to create a very limited system if it was going to run in 640K, so it would be really important if it were extensible to new sorts of things, and new function and new types of data. So, in some sense, that's true, I would never have phrased it that way perhaps, but yes!

What did you do instead?

We created Starburst, which was one of the first extensible database systems. It came up with a new way of optimizing queries, it's very extensible, rule-based, elegant (I think elegant, Guy Lohman did the work, I should be very clear on that!). But it was a system that would allow you to plug in new data storage systems, that would allow you to add new indexing methods, that would allow you to add new functions, one of the early user-defined functions capability, and still be able to optimize over these very different kinds of structures. Somewhere in there, the product division decided they didn't really need that much help to create their little PC database, which was fine by us, and so we got some space to do some real innovation, and to really rebuild from scratch, a second generation relational system. I guess what kind of put that work over the top was that it became the basis for IBM's DB2 for workstations (the DB2 that most people would know of and use today, not the mainframe system). To this day, some of the code we wrote in that project is part of that system.

Your career has certainly turned out well, considering that I heard that you were forced to go into the database field. Why the reluctance?

I had a really bad experience in grad school -- I'm old enough that relational databases were being invented when I was in graduate school. When I went to grad school, database was these network database systems, these complicated models, and we had a teacher who shall remain nameless, who just did not make it terrifically exciting. I didn't like the class at all. I declined to take my database qualifier because I was pretty sure I would flunk, because I was putting no effort into the class. So I was somewhat horrified when I got to IBM and they told me my first project would be a distributed database project. There had been a highly available systems project I thought was more interesting, but, in retrospect, I was very lucky, because that project didn't go, I think, as far, but it was doing the kinds of systemic and algorithmic things that I liked with distributed systems.

So you didn't do databases at Harvard with Phil Bernstein? Cause that was 100% relational…

No, I did not. I took his operating systems class, I took compilers… I decided very late that I wanted my minor, because at that point it was an applied mathematics major, in computer science, and so I didn't get as far into the curriculum as some people did. I spent a lot of time thinking I was going to major in linguistics, and in fact at Harvard, I got them to offer a seminar in Natural Language processing because I was so interested in languages and grammars and so on, and I knew people were doing that with computers and I wanted to know what was what. So I asked my advisor if there was any work in that at Harvard, and he said "no, but you know Tom Cheatham (who was the head of the department at the time) is really interested in that topic. Go talk to him. Maybe he will run a seminar". And they did, they did run a seminar and it was an all graduate seminar, and me. So I did a lot of other things and never made it to databases, it is sort of a shame. I did audit the database course when Umeshwar Dayal got to Texas and started teaching it, and it was so night and day different from what I had seen, that I really wished that I had spent a little more energy, and learned a little more. But no, I didn't intend to have anything to do with databases ever again.

You have spent most of your career in IBM research, but you also spent time on the products side at IBM. What should researchers know about product development?

I went into the product side as an escape. I had gotten myself too over-committed to all the things that were going on in my little industrial research world. At the time that I went there, I was transferring the work from Garlic into the product division. I had just started the Clio project, which was, I think, a fairly radically new, at the time, approach to do information integration, from the tooling side, not the infrastructure side and how we connect up different components. It was a cool research project. We were doing really great stuff, building prototypes, and writing papers, and at the same time, IBM had just launched a life sciences business, and they had decided that information integration, in particular the kind of federation of heterogeneous sources that Garlic had specialized in, were really key to getting a foothold in healthcare, in particular, the pharmaceutical industries. I found myself CTO of the life sciences business for IBM, in my spare time, while running this research project, and transferring my old one, and I was going nuts. I call those my guilt years. I had a young teenage son, and one in middle school, and they needed a lot of time and attention. So I was always a bad mommy; a bad researcher, because there would be a paper deadline, and I was talking to clients; a bad CTO because I should be on the phone with a client, but I'm writing my paper; I was always bad to somebody. So I went to development as an escape. I somehow just decided that there was no way to fix my life except to give up one or more of these roles; I just had to give it all up. And what I think database researchers should know is development is no escape! The pressures were, if anything, even more intense, there was a huge learning curve. I was Little Miss Purist Researcher. In fact, when I told people at IBM that I would be going down to development as a development manager, they were like: "You? That doesn't make any sense! You're the one who has had the least to do with development!". I was never hostile, but there were always people to buffer me, and I just didn't engage very well. So I learned more acronyms in a month than I had learned in my previous 20 years at IBM. I learned a lot of really cool things about how you get a product out the door, and all the work that goes on between what we in research think is a finished product, the cool prototype that we could write some new papers and run some great experiments on, and what customers really use. I learned a lot about how you have to support customers, and deal with customers. So I would tell people it is a really worthwhile experience to have, but don't do it just for the sake of being in development. Do it in something, an area you are kind of passionate about, pick your timing perfectly. I was really lucky. I went down there (it's the development "down there" because IBM Silicon Valley lab, our development lab, is 5 miles south of Almaden, and we're on the hill and they are on the flats), so I went down there to be the development manager for the query compiler for DB2, which was like going home to your grown-up child. So remember that Starburst code became the product basis, and it had grown up over the 10 years since then, or whatever, and so now it was a full-fledged product offering. It was kind of cool to be invited back to actually manage that team, and it gave me a certain amount of stature with the team, which was good since I didn't know anything, and I relied desperately on all the managers and people around to get me through. But the timing was perfect because, in fact, this group also owned the technology that we were transferring from Garlic, and they went on to launch our information integration business, and I got to be the development manager for this new business.

Why should one choose to become a manager?

I think being a manager gives you a very different perspective on the world in an industrial lab. It is kind of a growing up experience. It forces you. Some people do this naturally. But for me, at least, it really forced me to think about not just other people, and careers, in a way that I would have never thought about my own, but also to think about where the work had to go in developing a vision, to provide that kind of guidance and direction to people, and to help them understand. I don't ask a lot of questions of myself, I just kind of blindly do things, and so I think the thing about being a manager is other people asked me the questions that I should have been asking myself.

Like what?

Like, "how does this have anything to do with IBM's bottom line?", "why will anyone care?". Oh, that's an interesting question! "Ok, so what is our value proposition"? And then, I'm a smart enough person. Once I've been asked the right question, and I'm thinking about it, I can start to develop answers that I think are compelling to me and to others. But at least early in my career, I didn't have that ability to ask the right questions, now I do. Now I've learned what a lot of those right questions are.

What are some other ones, besides that one you have already mentioned?

Things like: "why does it have to be this way?". I get frustrated with procedures and processes and I'll ask about those myself. But, the more meaty questions of, "why do we have these kinds of rules?", and "why is it important that I join ACM?". I mean, I joined ACM as a student because my advisor told me to. I really had to think about: "well, okay, why did I stay a member of ACM?". We know this is a huge issue for ACM now, and now I can at least talk to people.

So, what is your answer?

I think you do it for your career. You can just show up at the conferences, you can just read the publications, but when you join something, you get a certain responsibility for it, and that kind of professional commitment causes you to change your behavior. I think if I had not joined, I would not have understood why I should be a vice president of SIGMOD at some point in the process. That was a wonderful experience, for my career, for networking, for understanding how things work in a broader world than IBM, and so on. There is a whole range of things. You don't do it for the publications; you really do it for your own career. I'm kind of a stickler for that. It is one of the things I lecture my team on. It doesn't have to be ACM, but some professional society, something outside of your company, or your university, or whatever. I think it is really important for your profession.

And in the part about questioning rules and procedures, do you mean you should have been more often questioning them, or more often accepting them?

I should have probably asked myself more questions. You know, somebody would say, distributed database (my first project at IBM), homogeneous database, "that's a good project", and so I just kind of worked on it. When we got to the second project, and they made me a manager, and they said we want this database for workstations, fortunately people in my group said "where is the research?", "what is it that we are doing?", and that was really what forced me on this path of, what are the key questions for us, and how are we going to answer them, and into this extensibility thing that we talked about. I was a shy, timid sort of person, and so for me, it was important just to grow up. People don't believe that now, that I was ever shy and quiet, but it was true. And I think even for people who do ask lots and lots of questions, I've seen it have almost the opposite effect: it forces them to kind of think about how what they say is interpreted in a different way. Because, as an individual, you are only speaking for an individual, and if you screw up, and somebody's mad at you, then that is your problem. If you are speaking for a team, and you didn't think before you spoke, and somebody decides that your project is crud, this can have a big impact on people. So, you get a level of responsibility that's very different. The other reason I always tell people they should be managers is because it is really good for parenthood, and vice versa. I learned a lot about being a parent from being a manager, I was a manager first, for about 8 months, I think. I took my first management job when I was pregnant (not sure I knew that at the time). So I had 8 months of practice, and you know, all the annoying little questions people will ask you, and how you have to teach people to think for themselves, and so on… When you go and apply that to children, it teaches you a lot of patience. I thought it was very useful.

So it sounds like it can help our readers in a lot of ways. So what sort of annoying little questions does one need to be ready to handle?

People think you are an instant expert on process and procedure. I am the worst person to ask about a process or procedure. I've learned to teach people to find their answers themselves. I teach them who my resources are. Who would go and do these things. It is hard to make sweeping generalizations, although I love to do that. But there is a class of people that are just very dependent, like the young child who doesn't want to let go of mommy's apron strings. They just ask mother-may-I way too often, and you need to kind of teach those people just like you teach a child that "you are responsible, you can move out from here". But that clinging… all the mothers know the clinging child that holds your apron string. Then there are the others that just cause trouble without thinking. They just speak without thinking, they offend somebody they really didn't have to offend, probably a teammate. It is like two children squabbling. You can just take any of the scenarios of parenthood, and they play out in the workplace, in a much more refined way. I mean, clearly, people don't punch or bite or scratch, but you get all the variants of behavior, which is why the analogy to parenthood works. I also found that it worked the other way: that being a parent taught me patience that I didn't have for grownups at the beginning. You get used to with your own children, to having to bite your tongue, and then think how to get them to do what they need to do, and that works well for the grownups too.

So thinking back, we have a lot of readers who are young people from distant lands, so some of them will be shy, and retiring. So thinking back to yourself all those years ago, what would you say to that person, from the vantage point where you are now?

I would say that too often people are afraid to try, when they offer us a position that we think we can't do, one that sounds like a really big stretch. Often they have seen something in us that we might see, but are afraid to admit to in ourselves, and we should just go with that. You should never take a position unless you really investigate it and you think you will enjoy it, and get something out of it. You should always take a position if you think it sounds like fun, but it is really scary. You know, it probably means you have ideas, and it probably means you are going to grow a lot out of it. So, it is hard for people, I know it was hard for me, when I was really young to think that I would ever enjoy being the leader. And I love it. I grew up into that. It took a long time, but I grew into it. And I think a lot of the readers, if they give themselves a chance, will too. So it is an experience I recommend. It doesn't have to be a manager, it could be a professor, you know, professors do this. It could be just a technical leader, leading some taskforce. That is really scary. It could be vice chair of SIGMOD or chair of SIGMOD, who knows, right? But they should give themselves the opportunity if people start to push them into it, even if to them it sounds like "wow, why would they think of me?". Especially, I'd say that to the women, and some of the folks in the global community, for whom it is just not natural to put yourself, as we would say in the States, "out there". Try it.

What about the reverse? In your management career, you have probably seen people who are naturally very bold, so we told the timid ones to be bolder, maybe push your comfort zone after investigating, and people may see things in you that you don't see in yourself. What about the people who are naturally very bold and want to lead everything? Do you have any words of advice for them when they are young?

Yes, learn to listen. Because, even if you are the most brilliant thing out there (and you may well be, I know some people like that), there are people around you who will have a different perspective, and who will teach you things, and make you even better and even more effective if you listen to them. What I often find in the brash young folks, is they are so busy impressing people that they don't necessarily listen and catch the subtleties of what people might be trying to tell them. So it is good to practice that if they are in a management or leadership position, professor, or whatever, they should practice taking a certain amount of time at each meeting to ask questions and then just "shush" and listen, because you need to get those skills. People tell you things if they think you will listen. If you aren't good at listening, you will lose that.

Do you have any words of advice for fledgling or midcareer database researchers or practitioners?

Enjoy. I just think this is the perfect moment to be in this field. I told the new researcher's symposium earlier this week that I am very, very bullish on the field right now. I see us poised to deal with all these vast data volumes, and different types of information. We have been promised for eons (it feels like) that we were entering the information age, right? We all believe we are there now, but we are still not getting all the value we should from that information. We are at the tip of the iceberg in terms of the impact that all this information is going to have. I think in the future everything is going to be about making all our processes and systems smarter: smarter energy, smarter health care, more individualized, less wasteful, all these different things. It is all about analyzing information, and then feeding that back in a control loop, and we can be at the heart of that if we want to. If we can understand what it is people want to do with the information, we can really pay attention to those applications, and engage to create the solutions that are needed. So I think it is just a really exciting time. We are moving from where I started, "oh, we are going to build some repositories for information, isn't that nice?". And now we are finally in an era where we can control the information and be the player that makes all this value come to pass from that data. So enjoy it. I think it is ours to really win and have a huge impact on the world. I am very bullish about that. I don't know another way to say it.

Among all your past research, do you have a favorite piece of work?

Yes, I do. Probably my favorite piece of work is the set of work Renee Miller and I started on something we called schema mapping. In the late 90's we had this notion that all the work that had been done on integrating data up to that point looked at what I would call the plumbing, the pipes, and how do you connect up the pipes to get data to flow to a particular place. But nobody had really thought about the design much, and how do you specify were those pipes should go, and what should happen to them, and so on. We set out to build some tools that would help people say how data should be transformed, and then we just applied database technology, especially the notion of having a higher level of abstraction and non-procedural, and all these good words we learned in the relational era, to the problem of information integration and ended up defining a new notion of schema mapping that would map from one schema and would say how one schema was related to another in a non-procedural way. Many people had talked about that before, but what we realized was that you could pose that problem in a way that could be viewed as query discovery, where you were really trying to find a query that related the schema you wanted to see with the schemas that you have. And if you could view it as a query, then you could compile it into many, many different runtime platforms. So this became a project called Clio, which started about 11 years ago, roughly, and wrote the first paper for VLDB 2000. So really the idea started to form crisply 10 years ago. I just really like that work because I think it is very elegant. It spawned a little subfield, at least for the theory community. There has been a lot of follow-on work that's happened, and I just feel very proud of it. I am very pleased with that piece of work. I think there is a lot more that needs to be done, so I am really glad other people have taken on some of these challenges, and are moving way past what we thought about. But for me, it was just a very nice elegant idea that turned out to be very practical. Some of the theoretical work led us develop algorithms that were very efficient for generating the transformation code, and then the ability to compile to multiple platforms such as federated database, so you could compile it to SQL, and then the federated database could execute it, or you could compile it to ETL scripts, or XPath and XSD, and a lot of XML transformation tools, or even to Java code. That was very powerful, it meant that you could express what your data looked like today, and what you wanted it to look like, and then depending on what a user's preferences were, you should be able to generate code to run on the right low level plumbing to meet the requirements of that user case.

What did people do before that work? They still had databases with different schemas before...

As I said, we had lots of plumbing, so they would make a decision: "I'm going to use a federated database system". They would then sit down and write by hand a bunch of SQL queries that would have to be debugged, and so on and so forth, that would in essence be the views that would transform the data. So it was a very manual process, and you weren't really saying how the data were related to each other, you were writing a query. It is just like writing code instead of writing the SQL query, only we were trying to move up another level in abstraction. Or they would say "I'm going to build a warehouse, so I am going to use one of these extract/transform/load ETL tools". And depending on which tool they chose, they would either write a little script (really write a program to extract data from one place, and put it to another), or the more sophisticated ones gave them these lovely dataflow tools where you could draw, you could just put things on a canvas and connect up the operators, and so on. It is still a program. What had spawned the Clio thought was my experiences in the product division, working with customers, and seeing how they would often like to use a federated database to prototype, just to see what they could do if they brought the data together, because they were much faster to set up. But the performance, if you were dealing with large data volumes, or doing these sorts of queries over and over again, it was not so good as if you built yourself a warehouse. Warehouse projects could take 6-12 months. They didn't want to go in there until they knew they wanted to do it. So they would set up a federated database, and then they'd ask, "so we think we understand what we want to get over here, but we have to build a warehouse. Is there some magic way to do that?". And there wasn't a magic way. I think what made Clio different was it offered a magic way: "well, you told us what the relationships were, and first we generated the federated queries, and we got your federated system working, and you got to see what you liked, and you could switch your mappings and regenerate the queries, and you could iterate much more quickly, and now when you are happy, you could generate out an ETL script and it could build your warehouse for you". That's cool :) We had a lot of success. In IBM at least, the Clio code has gone into a number of different IBM products, and so that is really nice as well. That is why that is my favorite work.

If you magically had enough extra time to do one additional thing at work that you are not doing now, what would it be?

This will be difficult to explain. I would be writing a newsletter. Because I have this worldwide role in addition to my day job running the computer science department at Almaden where I'm trying to help IBM's worldwide research team understand what it means to do research. I have a lot of young labs with very young researchers, some in economies where there is not quite the same tradition of scholarship or certainly of industrial research that we have. So I'm sort of team mom for the global team in terms of helping ensure that we have really shining cool projects and scientific examples going on worldwide, but also that the researchers grow and learn and so on. And so I feel a desperate need to communicate more with the global team. I can't be everywhere at once, unfortunately. I think a newsletter would help, but it takes a long time to write a newsletter, and I just don't have that kind of time.

If you could change one thing about yourself as a computer science researcher, what would it be?

I wish I had Mohan's ability to remember every article I ever read, and person I ever met, and how they are related, and who knows who. That would be really wonderful. And I wish I were more detail-oriented. I have to force myself to focus on the last, you know, crossing the T's and dotting the I's is not my strength. I am one of the people who would paint with a broad brush and wouldn't be so good at going back and filling in the details.

But when you are high up, you can't fill in the details, can you?

I know, but I think you could do better. I think I could be more disciplined. But I think it would be more useful for me to have that Mohan encyclopedia knowledge. That would be nice.

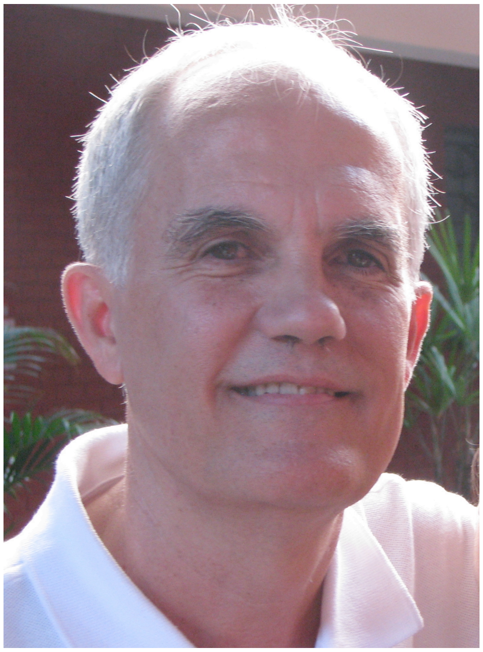
Thank you very much for talking with us today.

My pleasure.

Alberto Laender Speaks Out

on Why Google Bought His Startup, How to Evaluate Graduate Program Quality, How to Do High-Impact Research in a Developing Country, How Hyperinflation Nurtured Brazil's Software Industry, and More

by Marianne Winslett



Alberto Laender
http://www.ufmg.dcc.br/~laender

Welcome to this installment of ACM SIGMOD Record's series of interviews with distinguished members of the database community. I'm Marianne Winslett, and today we are at the SIGMOD 2010 conference in Indianapolis. I have here with me Alberto Laender, who has been a professor at the Federal University of Minas Gerais in Brazil since 1975. His research interests include web information systems, digital libraries, and data modeling. Alberto is a member of SIGMOD's Advisory Board and SIGMOD's Jim Gray PhD Dissertation Award Committee. Alberto's PhD is from the University of East Anglia. So, Alberto, welcome!

Please tell us about your startup company, Akwan.

Akwan was created in February of 2000. We had developed a search engine for Brazil, which was called *TodoBR,* and we started the company. We used the technology to provide service for the community, for industry, for companies. And then in July of 2005, Google bought the company. This was the beginning of the Google research and development center in Latin America, the first one. In fact, I believe that Akwan was the second company that Google bought outside the United States. The first one was from Canada or Australia, I'm not sure about that; and the second one was Akwan.

How is search with Akwan different than the normal Google search?

At that time, we had what is called a vertical search engine. It was completely dedicated to the Brazilian web. Because of that, we were able to do much better than Google at that time in Brazil. We knew about the Brazilian web, and everything was customized for the Portuguese language, and also the search engine was very fast too. And so it was the favorite search engine for the Brazilian users at that time. Nowadays, Google is there, and they have a much larger infrastructure than what we had at that time, so they can do better. They also hired all our engineers, so all the technology that we developed is now being used by Google Brazil.

Many database researchers have been lured into industry by the promise of access to real data. Since Google Brazil must have incredible access to web data, why aren't you working for them?

That was part of the deal. When they bought the company, as I said, all the engineers stayed with the company. They were former students from our university. Berthier Ribeiro, who was the CEO at that time, was the only Akwan founder that stayed with Google. I don't know, maybe it is some kind of policy from Google that they buy the companies, and they keep just the engineers, so that after that, they can try new ways of managing things, and new directions. We don't actually have a kind of agreement with Google at the moment, other than perhaps we have been helping them to hire new people in Brazil – that is the only thing that we do with them.

So you are not using the data from them in your research?

No, not really. This is a pity, because it would be interesting to have some kind of data from Google Brazil to know a little bit more about the Brazilian web nowadays.

You have experience in measuring the quality of CS departments, an area where Brazil fares very well. One aspect of that task is measuring the output and impact of individual researchers. In the rest of science and engineering, people like to use the Thomson ISI impact factors. But you and your colleagues concluded that "a look at the Thomson ISI list of the top 250 researchers with more citations in the CS field shows that this list includes none of the ACM Turing Award winners in the last 10 years, clear evidence that this database does not cover the relevant publication venues in the field" [Assessing the Research and Education Quality of the Top Brazilian Computer Science Graduate Programs, *inroads SIGCSE Bulletin* **40**(2), June 2008]. *What do you think is the best way to measure impact, for computer science researchers?*

That is a difficult question. I should mention that we got this example from a presentation from a researcher from Europe who was looking at the ISI impact factor. He brought in data saying that among the top researchers, he couldn't find any of the Turing Award people there.

In Brazil, we have a Ministry of Education agency, CAPES it is called, which is responsible for evaluating all graduate programs in Brazil from all areas. They create what we call the *Qualis*, a ranking of journals and conferences for all areas. This ranking divides all the venues into different levels, and this is used to assess the research produced by the graduate programs. Because they try to do this in a general way, so that it could be used for all areas, they have been using the ISI impact factor for some time. In the case of computer science, using the ISI impact factor is a problem because it is difficult to access the impact factor for conferences, and conferences are very important for computer scientists.

Also, compared with other fields like chemistry or even biology, the number of journals in computer science is not so large as in other areas, so sometimes it is really difficult to have all those papers and all those venues correctly classified when taking just into consideration the impact factor.

I don't have an answer for what would be the best way of assessing all the graduate programs. Clearly some kind of impact factor is important, but the ISI factor could not be seen as the right answer for this kind of evaluation for computer science.

Didn't you come up with a new way based on DBLP?

Yes. We used DBLP to do a comparison between Brazilian programs and programs from North America (the United States and Canada), and from Europe. The idea was to show that the Brazilian computer science programs had a good standard compared with major programs outside Brazil. So we took data from DBLP and we used the Qualis system for classifying the papers and the articles from all the researchers in these programs. Then we did a vertical comparison, where we compared each computer science graduate program in Brazil with the ones from the United States and the ones from Europe. I think this makes more sense than just trying to give some kind of grade for a program. The result was that the Brazilian programs are not as good as the top programs in the United States and Canada, like MIT or Berkeley or Stanford or Toronto, but we are in the middle. We did quite well in comparison with other programs in North America and also with programs in Europe. So that was the idea, to provide an assessment about how computer science was doing in Brazil, compared with well-established programs outside Brazil. The results of this study appeared in the *SIGCSE Bulletin*, and are available on line at http://portal.acm.org/citation.cfm?doid=1383602.1383654.

Why does Brazil produce so many database researchers?

That is a good question. I think I have two possibilities for answering this question. The first one is that when the graduate programs started in Brazil, databases was something very attractive, because it was a new area at the time. This was about 30 or 35 years ago. The relational model was just getting a lot of attention then. This attracted a lot of people. This I think was one point.

The second possibility was that the first graduate program in computer science in Brazil was the one from the Pontifícia Universidade Católica (PUC) in Rio. And there, Prof. Antonio Furtado, who is well known in the database community, started a very nice group and was responsible for graduating a lot of PhDs at that time. His former students started other database groups around Brazil. I think this was the main reason why the database community there was larger than the other ones.

As a consequence of this, the Brazilian Database Symposium was one of the first Brazilian computer science symposiums in Brazil. We are celebrating this year the 25th anniversary of SBBD, as we call it. SBBD nowadays is recognized as a very strong symposium in Brazil, and also it is seen outside Brazil as one of our major conferences.

If I look at the 2009 photo of SBBD attendees, I see so many female researchers. Why is that?

I really don't know, but this is true. The female community in the database area in Brazil is very large. It might be because when the database community started in Brazil, it was very common to have girls in the computer science undergraduate programs in Brazil. From some time, almost 50% of the students were female. But, nowadays it is not that high. I think it is the same problem here in the United States, that the number of women in the computer field has decreased very much. And it is like a tradition, we still have a lot of females working on databases in Brazil.

In the 1990s, Ricardo Baeza-Yates wrote that "except in the dubious domain of providing bodyshopping labor, it is proving very difficult for embryonic software industries in [lesser-developed countries] to be competitive in mature markets." Do you agree?

I can say that I don't agree, but I think that Ricardo is right in some sense. I think that in all developing countries, you can always find a way to do some kind of good research in computer science, and also transform these research results into some prototypes, and even have some startups as well.

In the case of Brazil, for instance, we have a very interesting situation now because the bank industry in Brazil has a very sophisticated computer system. The main reason for that was the fact that years ago we had a very high inflation in Brazil, and the banks had to be very efficient to process all the transactions.

Why does hyperinflation mean that banks have to process transactions fast?

With high inflation like Brazil had, if you just deposit a check, you cannot wait for one week to have your check cashed. You need the money almost immediately, otherwise the inflation will just, how can I say… your money can disappear in a few days, or at least lose a lot of its value. Because of this, all the bank financial systems had to be very efficient, and everything was automated.

The bank software that was developed in Brazil is nowadays being used in many other countries, in Europe, in Asia, and all over the world. And because the banking system and financial transfers had to be very secure, network security also is an area where Brazil has many companies who built software that is used outside Brazil today.

So the unique national circumstances of that hyperinflation became an opportunity for the local industry in Brazil. There is a saying about that: "If life gives you lemons, make lemonade."

Yes, exactly. That is the point.

What approach do you recommend for doing high-impact research while living in a developing country?

It is difficult to give a single answer for that question. I think that people have to look for good problems that are very interesting for their students, and also that can be used for many other people. For instance, I think the web is a great opportunity because the web is a very democratic environment. If you do some kind of research on the web in Brazil, or in Asia, or in some country in Africa, or here in the United States, what you do has to work for the whole web. If it works for the whole web, you have the opportunity to show that those ideas that you have are good for some kinds of applications, and that you can construct a useful system from there. You can have the opportunity to have some products, or some services, with the web as a basis. So you have a very nice environment to check and to test all the applications that you have developed. I think that the point is to find good problems where you can construct a prototype for their solution, based on the web. I think the web is a great opportunity for doing research in computer science in developing countries.

Can you give us examples of topics that maybe weren't so great to pursue in a developing country, where it is harder to have an impact?

For instance, trying to do something in computer architecture is almost impossible in a developing country because you need a whole industry behind you so that you can have impact in that. On the other hand, if you have a new kind of processor, you can do some research on how to use that processor for a specific application or how to improve, for instance, query processing in database systems; that you can do nicely. But trying to develop a whole new database management system is something that is hard work, and is very difficult to do in a developing country.

Tell us about your project where you are analyzing posts on Twitter and Facebook.

This is a project from our group, which is part of the National Institute of Science and Technology for the Web. It is a new kind of project that has been approved by the Brazilian government, and we are one of the four national institutes related to computer science in Brazil. This year is the presidential election in Brazil, and so we developed a whole framework to try to follow what is going on on the web in some specific subject. For instance, we have the elections in Brazil, we have the soccer World Cup in South Africa, and these are very sensitive subjects in Brazil. And so what we did was to follow Twitter, Facebook, all the things that happen on the web every day, and try to put all this data together so that the users could see what is going on in terms of the election, who is the candidate that has been exposed in the media, how the media is talking about the candidates, what was the discussion in terms of the major points for the election. Now we have the soccer World Cup in South Africa, and so we are doing more or less the same, following the players, the teams. Giving to the user an idea what is going on on the web about these subjects is very important, at least in Brazil. Soccer and elections are two subjects that are very important for Brazilians.

It sounds a little bit like another area for a startup company.

Sure, we do have a chance to do that. We have shown some results of this project to media companies, and they want to do the same thing for the brands of their clients. They want to show a customer how his company is seen on the web, and so we can use the same approach to help them. In fact, most of these media companies already do this, but they do it manually. They go to the web and try to collect data about a specific brand, or a specific company, and they have a lot of people doing this manually. Using some techniques from data mining, from classification, we can do at least a first step automatically, and give them a lot of results. Then they can use the people that they have behind the scene to check whether the data we collect from the web is okay. For them, it's much better doing it this way than trying to have people look at all the pages on the web, all the sites, to determine this feeling about the client's brand.

I know that other people in the database community are also working on sentiment analysis and that sort of thing. Is it different in Brazil, just like your search engine was different for Brazil? Is it customized in some way?

I don't think so, I don't think that it is different. It is interesting, for instance, that Brazil has the largest Orkut community in the world. Nobody knows why. Google is a major player there now, but when Orkut appeared suddenly in Brazil, all teenagers had to have their account in Orkut, and they use that to communicate among themselves. Brazilians like all the facilities available on the web today, and they are attracted to the things that happen on the web. People nowadays use all the social networks, and it is almost a must for the young people to have an account on Facebook or Orkut or any other social network.

Sounds like the US.

Yes.

Do you have any words of advice for fledgling or midcareer database researchers or practitioners?

The researcher must try to find a good research topic, and must learn about the work on that research topic that is going on in the world. They should try to cooperate with people who already have some experience on that topic. But particularly, in choosing the problem, it must be something you like. If you find something interesting, go after it. I think that is the main point that I would say to a young researcher, or people who are trying to start some kind of research in computer science in general. I think computer science is a fascinating area, and we have a lot of things to do in computer science.

Among all your past research, what is your favorite piece of work?

There are two works that I like best. The first one was when I started to work on traditional database modeling, and the second was when I used traditional database modeling techniques to help to develop a data extraction tool for web data.

I started my research career working on conceptual modeling. At that time the entity-relationship model was something that everybody used. I did a very nice piece of work together with my colleague from PUC Rio, Marco Casanova, and a PhD student of mine at the time. We developed some techniques for mapping from the entity-relationship model to the relational model, in such a way that you could correctly represent everything that was expressed in that ER diagram. It was interesting because it was a kind of logical database design method. The results that we got at the time were very useful in practical situations. We even implemented some tools, which is something that I like very much to do. When I returned from my PhD, this was my first research project, so it was very important for me.

Much later on, I was doing quite traditional database research, and I wanted to change. The web already existed, but we didn't know very much about the web. I had the opportunity to spend a few weeks at HP Labs in Palo Alto. There I worked with Moshe Zloof and started trying to do something somewhat related to Query By Example. Moshe had a project that was called the Picture Project, where they were trying to use the same paradigm as Query By Example, but also for creating interfaces and creating and developing software by example. On the web at that time, you found a lot of data, basically HTML pages from many different sources. I had the idea that we could extract that data so that we could store it in a database and develop some traditional applications. I had the idea to use the paradigm of programming by example for this task, and we created DEByE, which is Data Extraction By Example. This became the major topic of a PhD student of mine. We developed a tool that was one of the first data extraction tools that could extract data from web pages that had some kind of internal structure. Most of the tools at that time were able to extract only tuples, and we were able to extract data with a hierarchical structure, so it was a very powerful data extraction tool at that time. Our results appeared in a survey that we wrote for *SIGMOD Record* on data extraction [A. Laender, B. Ribeiro-Neto, A. Soares da Silva, J. Teixeira, A Brief Survey of Web Data Extraction Tools, *SIGMOD Record* **31**(2), June 2002; http://www.sigmod.org/publications/sigmod-record/0206/laender-survey.pdf]. We had another paper on that topic in *Data and Knowledge Engineering* as well [A. Laender, B. Ribeiro-Neto, A. Soares da Silva, DEByE - Data Extraction By Example, *DKE* 40(2), 2002, http://www.sciencedirect.com/science/article/pii/S0169023X01000477].

Later on, we had a paper in the WWW conference [D. de Castro Reis, P. Golgher, A. Soares da Silva, A. Laender, Automatic web news extraction using tree edit distance, WWW 2004], for which we developed another tool for extraction of news from the web. We used this extraction technique at Akwan for developing clipping systems for news stories, which extracted the headlines of news from the web. Using that, you could provide for companies a summary of the news from all the newspapers and the web sites that we had at that time. This paper describes a very interesting technique for extracting a particular kind of data from the web, news. You have to recognize exactly all the specific parts of a web page that you want to extract.

If you magically had enough extra time to do one additional thing at work that you are not doing now, what would it be?

We always want to do a lot of new things, but I would like to have more time to stay with the students at the labs, and maybe do some programming. I liked very much to write programs when I was doing my PhD studies. After you come back and start your own group, you have to manage all the students and you have to write project proposals to get some grants, and the end result is that you are very far from the activities in the labs. So I would like to have more time to be there. I had some opportunity for doing that when we started our company, because I stayed there for almost 6 months working with the engineers on some projects, and this was very nice. But I would like to do this again with the students. Maybe when I retire, I can come back to the lab to do some programming with them.

If you could change one thing about yourself as a computer science researcher, what would it be?

This is a difficult question. After so many years, it is difficult to see what I should have done.

I decided to work on databases. I think that if I had enough time, I would like to go to see a little bit more about other areas, like computer networks, for instance. With the web today, I would like to be able to understand a little bit more about the whole aspects of the network, the physical aspects of the network, but it is almost impossible to do this with the responsibilities that I have now. I think that sometimes maybe when you start doing your research, you dedicate too much time only to that subject, and there are other areas that are so interesting. But it is difficult to look at more than one area at the same time.

Thank you very much for talking with me today.

It was my pleasure.

Report on the First International Workshop on Flash-Based Database Systems (FlashDB 2011)

Xiaofeng Meng†, Peiquan Jin‡, Wei Cao†, Lihua Yue‡

† School of Information, Renmin University of China, Beijing, China
‡ University of Science and Technology of China, Hefei, China

1. INTRODUCTION

Recently, new storage media such as flash memory have been developed very quickly, which brings big challenges to the architecture of computer systems as well as the design of system software. In particular, NAND flash (either SLC- or MLC-based) in the form of solid state disks (SSDs) has been an alternative to traditional magnetic disks, both in the home-user environment and in the enterprise computing environment, due to its shock-resistance, low power consumption, non-volatile, and high I/O speed [1]. The special features of flash memory and other new storage media impose new challenges to traditional data management technologies. As a result, traditional database architectures and algorithms designed for magnetic-disk-based storage fail to utilize new storage media efficiently. Meanwhile, the new characteristics of modern storage media, such as not-in-place update and asymmetric read/write/erase latencies of flash memory, also bring great challenges in optimizing database performance, by using new querying algorithms [2], indexes [3], buffer management schemes [4], and new transaction processing protocols. Consequently, exploiting the characteristics of flash memory and other new storage media has become an important topic of database systems research.

In order to make database systems adapt automatically to the characteristics of flash memory and other new storage media, the data management community needs to rethink traditional underlying storage architecture, query processing algorithms, indexing mechanism, buffer management schemes as well as many traditional issues in magnetic-disk-oriented database systems to adapt to the advances in the underlying storage infrastructure.

The First International Workshop on Flash-based Database Systems (FlashDB 2011) was held in conjunction with DASFAA 2011 in Hong Kong on April 22. This full-day event brought together researchers and engineers from academia and industry to discuss and exchange ideas related to flash-based database technologies. The workshop features three invited talks and two research sessions. This summary report gives a concise view of the three invited talks, as well as the novel ideas presented and discussed at the workshop. We hope this report will help the community by conveying the inspiring ideas and topics which form the frontier of this research area.

The workshop began with the opening speech given by Xiaofeng Meng, professor from Renmin University of China. This workshop was co-organized by researchers from Renmin University, University of Science and Technology of China, and Hong Kong Baptist University, and attracted up to twenty attendees from Korea, Germany, France, mainland China, and Hong Kong China in academia as well as in industry. Flash memory has been growing as a new type of storage media with its advantages such as faster IO speed, lower power consumption, better shock resistance etc. compared to magnetic disks. At present, it is still an open issue to utilize the advantages of flash memory to achieve better system performance and higher energy efficiency in current database systems. This workshop aims at serving as a platform to share and exchange ideas, to work together to address flash-based database related problems, and to nurture inspirations of new solutions in this area.

2. INVITED TALKS

The first invited talk was given by Sang-Won Lee, professor at Sungkyunkwan University. The talk was titled as “Some Research Directions in FlashDB” and had four topics. Firstly, the speaker reflected on transactional in-page logging (TIPL) for multi-version read consistency and recovery, the transactional support of in-page logging (IPL[5]) design on NAND flash memory that employs out-of-place update and fast read speed of flash memory. TIPL takes advantage of redo logs dwelt within blocks offering multi-version store and new recovery schemes with nominal overhead. Performance evaluation from event-driven simulators of TPCC traces of multi-version read consistency and fast recovery shows the effectiveness of TIPL. The second topic began with threats and opportunities IPL design faced with the development of flash memory and the emergence of PRAM (i.e., Phase Change Memory, PCM). With this vision, IPL-P (IPL with PRAM) was proposed as a hybrid storage design based on flash memory and PRAM, to keep page-oriented logs on PRAM to utilize the better small-sized-write efficiency of PRAM. IPL-P outperforms flash-only design in simulated evaluation and in real board evaluation outperforms both flash-only and PRAM-only designs for insert and update operations. In the third topic, Prof. Lee discussed the design consideration of FlashCache, which uses flash-based SSD as extended buffer cache of RAM in the hybrid storage architecture involving both HDD and SSD. He also presented the improved performance when running the TPC-C benchmark on PostgreSQL. Finally, the speaker retrospected the concept of DB machines in the light of SSDs. The breakthrough in flash read interface and the parallelism inside SSDs, data-intensive computing and "bandwidth crisis" confront the host CPU with more burdens. Inspired by current SoC technology such as hardware-based ISP, Prof. Lee rethinks realizing some database computations, e.g. scans, aggregation, joins, and sorting, on embedded CPUs of storage devices to offload the host CPU, a shift from "bring data to computation" to "bring computation to data".

The second invited talk was given by Theo Härder, professor at the University of Kaiserslautern. The title of his presentation was "Energy Efficiency is not Enough, Energy Proportionality is Needed!"[6]. His talk included four parts. The first part described the characteristics of flash memory and SSDs and showed the differences of different SSD types on the basis of empirical experimental results. A number of issues were explored including whether SSDs suffer from random access, whether SSDs exhibit unstable and fluctuating behavior, whether read/write asymmetry is as bad as commonly expected, whether overwriting of blocks on a full disk is much slower than writing to an empty disk, and whether queue depth has an impact on performance. Energy-consuming experiments have revealed that different SSDs have different power profiles and that the power consumption for idle states and peak loads is considerably lower than for HDD. A critical question concerning energy consumption is whether energy efficiency and energy proportionality observed at the SSD device level can be also expected at the system level. For this reason, the second part compared disk- and SSD-based DBMS buffer management methods, such as CFDC, CFLRU, LRU, LRU-WSR and REF. The CFDC algorithm was generally superior to its competitor algorithms w. r. t. performance and energy efficiency. However, the energy consumption of ATX-, IDE- with a SATA-based disk at the different processing states such as idle, working and peak, revealed that these components are not energy proportional to system utilization. In the third part, Prof. Härder analyzed the relationship between the power use and the system utilization, including CPU, hard disk and SSD, and discussed how energy-proportional computing could be achieved. Ideal energy-proportional computing should consume no energy in idle states; power consumption should linearly increase with system utilization and approach full power usage (100%) in peak load situations (100% system utilization). However, current computer systems are not energy proportional at all, because major components (main memory, parts of ATX, etc.) consume an almost constant amount of energy independent of the degree of system utilization. Therefore, the entire system reaches at idle states often already more than 50% (in case of a very large main memory close to 100%) of the energy consumption needed for peak load. The last part introduced a research project aiming at energy-proportional computing in the context of DBMS use. The system called WattDB uses an architecture where the powerful DB server machine is replaced by a cluster of wimpy shared-nothing computing nodes and some shared-disk storage nodes. By activating the processing nodes on demand, power consumption of the entire system can be decreased to a minimum level and, thus, energy proportionality can be approximated. Starting with a single node in the cluster, additional nodes can be activated on demand without interrupting DB processing. In this way, the cluster is able to scale up to n nodes and smoothly grow and shrink, so WattDB can stepwise approximate an ideal energy-proportional behavior. Each of the individual computing nodes is able to access the entire database via storage nodes. Because of dynamic node fluctuation, frequent DB cluster coordination is necessary to optimally support DB processing and maintenance as well as concurrency control and logging/recovery, DB partitioning, etc. Methods for flexible physiological DB partitioning have to be developed to successfully reach the research objectives of WattDB. Finally, Prof. Härder stated that "In the future, WattDB will be specialized towards differing directions to provide tailor-made support for the application classes OLTP, OLAP, and MapReduce".

Jianliang Xu, professor at Hong Kong Baptist University, was the third invited speaker. In this talk entitled "Flash-based Database Systems: Some Experiences from the FlashDB Project", Prof. Xu first introduced the FlashDB project, an NSFC key project collaboratively carried out by three institutions in mainland China and Hong Kong. The goal of the project is to investigate new architectures and methods to boost database performance, by exploiting unique flash I/O characteristics. The speaker exemplified three case studies towards this goal. In the first case study, DigestJoin is a two-phase flash based join processing method that makes good use of random reads on flash memory devices and reduces writing of intermediate join results [7]. In the first phase, digest tables in the form of <join_key, tuple_id> are generated and then joined. In the second phase, based on the digest join results, full tuples are reloaded to form the final join results. The second case study addressed a new approach for write performance optimization [8]. Based on the observations that sequential, focused, and partitioned writes are more efficient than general random ones on flash-based storage devices, by setting up a small sized (e.g. 1–16 MB) stable buffer on flash devices, a general random write can be transformed into a focused write to the stable buffer and an efficient flush of pages from the stable buffer to the destination. And the third case study presented that the shadow paging technique well suits out-of-place updates of flash memory devices. Combined with the partial page programming feature in SLC flash memory, a novel flag commit idea was discussed to support transaction recovery. Two specific protocols, Commit-based Flag Commit (CFC) and Abort-based Flag Commit (AFC), were designed to support normal transaction processing, commit/abort, garbage collection, and recovery.

3. RESEARCH SESSIONS

3.1 Session A: Storage Management for SSD

The fundamental thing to successfully adopt flash devices in database systems is a storage design that is based on the specific storage and access characteristics of flash memory or flash devices. This session features four research papers addressing this problem from different aspects.

The paper entitled Page-Level Log Mapping: From Many-to-Many Mapping to One-to-One Mapping addressed the logical-to-physical page mapping issue in flash-based systems. The authors designed a page-level log mapping method called PLM, which uses backward link technique to support efficient reads and writes, and therefore can yield optimal overall performance. Besides, the authors developed two implementations of PLM incorporating flash-optimized strategies for buffer management, free page allocation and garbage collection. Finally, the proposed algorithm achieved high efficiency across a series of experiments.

In the paper entitled A Novel Method to Extend Memory Lifetime in Flash-based DBMS the authors first analyzed the previous methods for free space management in conventional DBMSs, such as free list and space map, and pointed out that those traditional approaches are not suitable for flash-based database systems. Therefore, the authors proposed to use an Append-Only (AO) scheme to maintain the free space in DBMS. The AO scheme allocates new empty pages as soon as a write request comes and appends it to the tail of the original database files, which avoids useless searching for a page with free space. Furthermore, in order to reduce the number of small write and random write, a stand-alone write buffer was also proposed to collect the inserted and updated records. The experiment result based on a flexible emulator showed this approach enjoys a 74.5% page write decrease.

The paper entitled Log-Compact R-Tree: an Efficient Spatial Index for SSD proposed a novel flash-aware variant of R-Tree, named LCR-Tree, which records the updates of R-Tree as logs to transfer random writes to sequential ones. Distinguished from previous attempts, compacted log was introduced to combine newly arrival logs with the original ones on the same node, which renders great decrement of random writes with at most one additional read for each node access. In this way, although more read overhead is invoked, the write performance is improved significantly. The experimental results on both synthetic and real data sets showed that the LCR-Tree can achieve up to 3X gains over RFTL, an existing flash-based index scheme, and the R-tree

The paper entitled An FTL- agnostic Layer to Improve Random Write on Flash Memory proposed a data placement algorithm specially designed for flash memory to improve the efficiency of random writes. In this paper, the authors first claimed that there is a strong correlation between write performances and spatial locality for FTL-based flash devices, and defined a distance between logical pages to reflect this effect. Based on the concept of page distance, the authors proposed a simple data placement algorithm which aims at transforming random writes into quasi-sequential access patterns trades. The efficiency of such a mechanism was validated by a formal mathematical model. In the experiment, the proposed method improved the random write performance by up to two orders of magnitude.

3.2 Session B: Energy Efficiency & Hybrid Storage

Energy efficiency is one of the key merits of flash memory. How to design a system that keeps high performance while saving energy is a challenging problem. The first presented paper addressed this issue. The other two presented papers discussed hybrid storage architecture of NAND flash and PRAM memory on mobile devices and hybrid storage of HDD and SSD.

The paper entitled Trading Memory for Performance and Energy mainly addressed the problem of tradeoff between performance and power consumption when managing extremely large amounts of data. From the standpoint of architecture, a three-layer database storage system was designed and implemented for reducing the power consumption. The prototype uses flash-based devices as an intermediate caching layer. The memory and disk layers are basically the same as those in the classical two-layer disk-based storage systems. To manage the flash layer, two algorithms, namely the Local (LOC) algorithm and the Global (GLB) algorithm, were presented as the replacement policy. Both experiments on synthetic and real-life traces were conducted to measure the overall performance and energy efficiency. The results showed that flash-based layer significantly improves the I/O efficiency and then reduces the use of energy-inefficient RAM-based memory without compromising the overall system performance.

The paper entitled Design of embedded database based on hybrid storage of PRAM and NAND flash memory studied the problems of database systems on mobile devices with a single storage media – NAND flash and a single file system, YAFFS2. To overcome the inefficiency of small-sized data read/write operations and frequent updates, hybrid storage architecture of PRAM and NAND flash memory was proposed to take advantage of the specific properties of PRAM, i.e., byte addressability and in-place updates. The proposed architecture replaces NOR flash memory in the conventional architecture by using PRAM memory as boot-up code storage as well as a data storage. Such a hybrid system was implemented on the basis of SQLite and dual file systems (YAFFS2 and PRAMFS). Particularly, the rollback journals of SQLite are stored on PRAM via the file system PRAMFS, while the database files are stored on NAND flash memory through the file system YAFFS2. Evaluation on board with NAND flash and PRAM emulated by UtRAM [9] with software delay showed the proposed architecture reduces the transaction time by 45% compared with systems equipped with only NAND flash memory.

In the paper entitled Hybrid Storage with Disk Based Write Cache, the authors proposed using HDDs as the write cache for SSDs to exert the better sequential write performance of HDDs while avoiding random writes on SSDs. In this hybrid storage architecture, pages are read from both HDD and SSD, while updated pages are all written to HDD once evicted from buffer. To take advantage of the high read speed of SSDs, the authors presented an approach to migrating read-mostly pages into SSD, in case that they are first located in HDD. Those migrated pages are organized as blocks and all migrations are performed according to a block unit, which aims at making use of the high sequential-write performance of SSD and also reducing the erase times of flash memory. Experiments were performed on several synthetic traces, and the results showed that the hybrid scheme ensures most read operations are performed on SSD and most write operations are focused on HDD. Meanwhile, it has less runtime than the single-disk-based mechanism.

4. DISCUSSIONS

At the end of the workshop, some open questions were identified by Prof. Jianliang Xu in his presentation concerning some potential research directions, which are summarized as follows:

(a) What are the main challenges and issues for flash-based enterprise database applications? While SSDs have been adopted as an alternative storage for enterprise database applications, architectures, data structures, and algorithms optimized for such applications should be developed in accordance with the performance objective such as information access speed, energy efficiency, or even endurance of flash devices.

(b) Which storage hierarchy will prevail in the future? With the advent of flash memory and the coexistence of magnetic disks, will flash memory serve as an extension of main memory or an extension of magnetic disks? Many schemes for hybrid storage have been proposed; but which one will prevail in the future remains to be seen.

(c) What is the impact of new NVRAM storage technologies such as Phase Change Memory (PCM)? PCM emerges with better I/O bandwidth, longer write endurance, bit-alterability, and byte-addressability, compared to flash memory. How to utilize the advantageous features of NVRAM to complement current storage systems is a very promising and interesting research problem.

5. ACKNOWLEDGEMENTS

This work was supported by the National Natural Science Foundation of China under the grant No. 60833005 and No. 61073039.

6. REFERENCES

[1] J. Gray and B. Fitzgerald, Flash Disk Opportunity for Server Applications, ACM Queue, vol. 6(4), pp.18–23, 2008.

[2] D. Tsirogiannis, S. Harizopoulos, M. Shah, J. Wiener, Goetz Graefe. Query Processing Techniques for Solid State Drives, In Proc. of SIGMOD'09, pp. 59-72, 2009.

[3] S. Yin, P. Pucheral, X. Meng, PBFilter: Indexing Flash-Resident Data Through Partitioned Summaries, In Proc. of CIKM'08, pp. 1333-1334, 2008.

[4] Z. Li, P. Jin, X. Su, K. Cui, L. Yue, CCF-LRU: A New Buffer Replacement Algorithm for Flash Memory, IEEE Trans. on Consumer Electronics, Vol.55(3), pp.1351-1359, 2009.

[5] S. –W. Lee, B. Moon, Design of flash-based DBMS: an In-Page Logging Approach, In Proc. of SIGMOD'07, pp. 55-66, 2007.

[6] T. Härder, V. Hudlet, Y. Ou, D. Schall, Energy Efficiency is not Enough, Energy Proportionality is Needed!, In Proc. of DASFAA'11 Workshops, LNCS 6637, pp. 226-239, 2011.

[7] Y. Li, S. T. On, J. Xu, B. Choi, H. Hu, DigestJoin: Exploiting Fast Random Reads for Flash-based Joins, In Proc. of MDM'09, pp. 152-161, 2009

[8] Y. Li, J. Xu, B. Choi, H. Hu, StableBuffer : Optimizing Write Performance for DBMS Applications on Flash Devices, In Proc. of CIKM'10, pp. 339-348, 2010

[9] Y. Park, S.-H. Lim, C. Lee, K. H. Park, PFFS: A Scalable Flash Memory File System for the Hybrid Architecture of Phase-Change RAM, In Proc. of SAC '08, pp. 1498–1503, 2008

Repeatability and Workability Evaluation of SIGMOD 2011

Philippe Bonnet[1], Stefan Manegold[2], Matias Bjørling[1], Wei Cao[3], Javier Gonzalez[1], Joel Granados[1], Nancy Hall[4], Stratos Idreos[2], Milena Ivanova[2], Ryan Johnson[5], David Koop[6], Tim Kraska[7], René Müller[8], Dan Olteanu[9], Paolo Papotti[10], Christine Reilly[11], Dimitris Tsirogiannis[12], Cong Yu[13], Juliana Freire[6], and Dennis Shasha[14]

[1]ITU, Denmark
[2]CWI, Netherlands
[3]Remnin University, China
[4]University of Wisconsin, USA
[5]University of Toronto, Canada
[6]University of Utah, USA
[7]UC Berkeley,USA
[8]IBM Almaden, USA
[9]Oxford University, UK
[10]Università Roma Tre, Italy
[11]University of Texas Pan Am, USA
[12]Microsoft, USA
[13]Google, USA
[14]New York University, USA

ABSTRACT

SIGMOD has offered, since 2008, to verify the experiments published in the papers accepted at the conference. This year, we have been in charge of reproducing the experiments provided by the authors (repeatability), and exploring changes to experiment parameters (workability). In this paper, we assess the SIGMOD repeatability process in terms of participation, review process and results. While the participation is stable in terms of number of submissions, we find this year a sharp contrast between the high participation from Asian authors and the low participation from American authors. We also find that most experiments are distributed as Linux packages accompanied by instructions on how to setup and run the experiments. We are still far from the vision of executable papers.

1. INTRODUCTION

The assessments of the repeatability process conducted in 2008 and 2009 pointed out several problems linked with reviewing experimental work [2, 3]. There are obvious barriers to sharing the data and software needed to repeat experiments (e.g., private data sets, IP/licensing issues, specific hardware). Setting up and running experiments requires a lot of time and work. Last but not least, repeating an experiment does not guarantee its correctness or relevance.

So, why bother? We think that the repeatability process is important because it is *good scientific practise.* To quote the guidelines for research integrity and good scientific practice adopted by ETH Zurich[1]: All steps in the treatment of primary data must be documented *in a form appropriate to the discipline in question* in such a way as to ensure that the results obtained from the primary data can be reproduced completely.

The repeatability process is based on the idea that in our discipline, the most appropriate way to document the treatment of primary data is to ensure that either (a) the computational processes that lead to the generation of primary data can be reproduced and/or (b) the computational processes that execute on primary data can be repeated and possibly extended. Obviously, the primary data obtained from a long measurement campaign cannot be reproduced. But our take is that the best way to document the treatment of these primary data is to publish the computational processes that have been used to derive relevant graphs. On the other hand, the primary data obtained when analyzing the performance of a self-contained software component should be reproducible. Ultimately, a reviewer or a reader should be able to re-execute and possibly modify the computational processes that led to a given graph. This vision of executable papers has been articulated in [1].

This year, as a first step towards executable papers, we encouraged SIGMOD authors to adhere to the fol-

---

[1] `http://www.vpf.ethz.ch/services/researchethics/Broschure`

lowing guidelines:[2]

(a) Use a virtual machine (VM) as the environment for experiments.

(b) Explicitly represent pre- and post-conditions for setup and execution tasks.

(c) Rely on a provenance-based workflow infrastructure to automate experimental setup and execution tasks.

Ideally, a common infrastructure guarantees the uniformity of representation across experiments so reviewers need not re-learn the experimental setup for each submission. The structure of workflows should help reviewers understand the design of the experiments as well as determine which portions of the code are accessible. While virtual machines ensure the portability of the experiments so reviewers need not worry about system inconsistencies, explicit pre- and post-conditions make it possible for reviewers to check the correctness of the experiment under the given conditions.

In the rest of the paper, we look back on the repeatability process conducted for SIGMOD 2011.

2. ASSESSMENT

2.1 Participation

Renée Miller, PC-chair for SIGMOD 2011, agreed to add a couple of questions to the submission site. 73% of the authors said that they would participate in the repeatability process. As we will see in Section 2.1.2, the percentage of accepted papers actually submitted to the repeatability and workability committee was limited to 35%. The reasons cited for not participating were:

1. intellectual property rights on software
2. sensitive data
3. specific hardware requirements

None of these reasons, however, explain the geographic distribution of authors participating to the repeatability process shown in Figure 1. This graph compares the number of papers accepted at SIGMOD and the number of papers participating in the repeatability process grouped by the region of origin of the first author (Asia, America, Europe, Oceania). While this grouping is largely arbitrary (some authors might not be associated to the same region as the first author), the trends that appears in Figure 1 is significant. To put it bluntly, almost all Asian authors participate in the repeatability process, while few American authors do. Some American authors have complained that the process requires too much work for the benefit derived [2], but we believe that several observations can improve this cost/benefit calculation

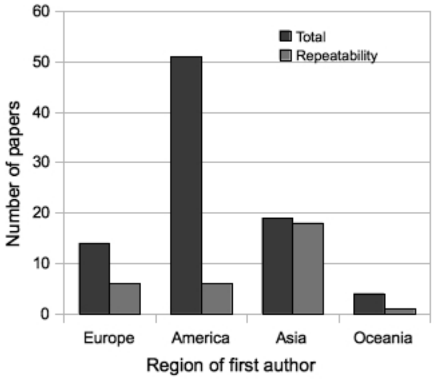


Figure 1: Distribution of participants to the repeatability process per region of first author.

1. **[more benefit]** repeatable and workable experiments bring several benefits to a research group besides an objective seal of quality: a) higher quality software resulting from the discipline of building repeatable code b) an improved ability to train newcomers to a project by having them "play with the system"
2. **[less cost]** using the right tools, a research group can make a research experiment repeatable easily (we are working on an upcoming companion article which contains a tutorial on how to make this happen).

2.1.1 *Process*

As in 2009, our goal was to complete the repeatability reviews before the conference started, so that authors could advertise their result during their presentation (a first straightforward way to guarantee some benefit for authors). We placed the submission to the repeatability committee at the same time as the deadline for the camera ready copy of the paper: leaving one month to the author of accepted papers to prepare their submission and leaving two months for reviewers to work on an average of three submissions each.

The availability of the Elastic Cloud Computing infrastructure via a grant from Amazon allowed us to experiment with a great variety of hardware and software platforms. Experiments were run on servers equipped with 26 CPUs or 40 GB of RAM, running OS ranging from Windows to CentOS. The availability of the Condor-based Batlab infrastructure from Miron Livny's group at U.Wisconsin allowed a reviewer to repeat a

---

[2] See the Repeatability section of the ACM SIGMOD 2011 home page: `http://www.sigmod2011.org/calls_papers_sigmod_research_repeatability.shtml`

cluster-based experiment with 40 machines - as opposed to 3 on the original paper. Note also that a few authors made their own cluster infrastructure available via a gateway which made it possible for reviewers to repeat the data acquisition phase of the authors' papers.

The most frequently asked question by authors at submission time was *where can I upload a package with the system and data needed to reproduce my experiments?*. Authors were asked to make their experiment available for download. This was a major problem for a Chinese group whose experiment package could not be downloaded properly despite numerous attempts. On the other hand, a group from ETH Zurich fully complied to ETH *Guidelines for research integrity* and made their software and data publicly available online[3].

A problem mentioned in the previous editions of the repeatability process was the high burden on reviewers when setting up experiments. To mitigate this problem, as explained in the introduction, we advocated this year that authors should consider submitting a virtual machine containing system and data. This effort was far from successful as illustrated in Figure 2. The vast majority of submissions were Linux or Windows packages with instructions on how to set them up and run the experiments. For most papers, the set up phase (specially on Linux) was well designed and required low overhead for the reviewer. However, many papers which did not get the repeatability label failed in the set-up phase, often because some dependencies had not been made explcit; such problems would have been avoided with a well tested virtual machine.

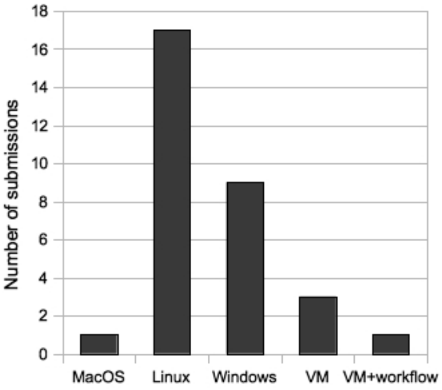


Figure 2: Operating system used for the submissions to the repeatability process in 2011.

Each paper was assigned a primary reviewer. A secondary reviewer was introduced in case the primary reviewers had problems downloading a submission, or setting it up because of OS or hardware mismatch. The load on the reviewers was quite uneven. Figure 3 shows the number of experiments per paper - which is a good indicator of the time needed to run the experiments. We still miss a good indicator for the time needed to setup the experiments. This year, we simplified the grades given to each paper: not repeatable, repeatable or repeatable&workable.

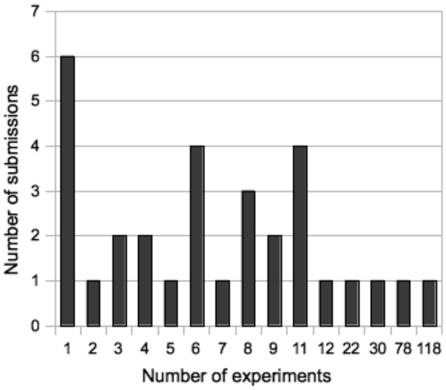


Figure 3: Distribution of number of experiments per submission.

This year, we set up a web site running on an EC2 server www.sigmod11repeatability.org with instructions for authors, a couple of examples showing how to use the Vistrails workflow engine to setup experiments and the submission site. We relied on an instance of HotCRP[4] to support submissions of instructions as well as anonymous interactions between authors and reviewers during the reviewing period. While HotCRP was fully satisfactory in terms of stability, functionality and ease of use; the setting of automatic emails from a GMail account created for the `sigmod11repeatability.org` domain turned out to be a problem - spam filters prevented mails and notifications sent by hotCRP to reach their destination.

2.1.2 *Results*

Figure 4 shows the results from the repeatability process since 2008[5]. In terms of percentages, the participation increased slighlty in 2011 compared to 2009 and 2010–those years where only accepted papers were considered for the repeatability process–while the percentage of repeatable papers remained stable.

The results were announced to the authors prior to the conference (at the exception of two papers). Results will be associated as labels on the existing article repositories

---

[3] `http://people.inf.ethz.ch/jteubner/publications/soccer-players/`

[4] `http://www.cs.ucla.edu/~kohler/hotcrp/`

[5] The results from 2008 and 2009 are presented in the SIGMOD Record articles [2, 3]; the results from 2010 are available at `http://event.cwi.nl/SIGMOD-RWE/2010/`

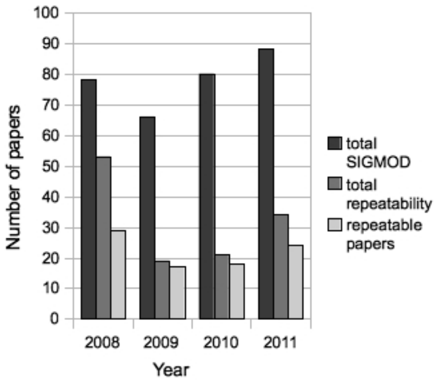


Figure 4: Repeatability results since 2008

(either ACM or PubZone). More importantly, the experiments themselves should be archived in a repository of repeatable experiment. Setting up such a repository for the SIGMOD community is the next obvious challenge.

3. CONCLUSION

The SIGMOD 2011 repeatability initiative attempted to increase participation and enhance the quality of submissions by offering tools to authors and reviewers. This has succeeded only partly: virtual machines and workflows simplify the process for reviewers but are harder to implement for authors than sending a shell script. Unfortunately, the shell scripts have many system dependencies that may make them difficult to repeat or to build upon by future researchers. An ongoing research challenge is to develop tools to help authors create high quality repeatable computational experiments with reasonable effort.

Acknowledgements

We would like to thank Amazon for their EC2 grant and Miron Livny and his team (especially Brooklin Gore) for their help with the Batlab infrastructure.

4. REFERENCES

[1] David Koop, Emanuele Santos, Phillip Mates, Huy T. Vo, Philippe Bonnet, Bela Bauer, Brigitte Surer, Matthias Troyer, Dean N. Williams, Joel E. Tohline, Juliana Freire, and Cláudio T. Silva. A provenance-based infrastructure to support the life cycle of executable papers. *Procedia CS*, 4:648–657, 2011.

[2] S. Manegold, I. Manolescu, L. Afanasiev, J. Feng, G. Gou, M. Hadjieleftheriou, S. Harizopoulos, P. Kalnis, K. Karanasos, D. Laurent, M. Lupu, N. Onose, C. Ré, V. Sans, P. Senellart, T. Wu, and D. Shasha. Repeatability & workability evaluation of SIGMOD 2009. *SIGMOD Rec.*, 38:40–43, December 2010.

[3] I. Manolescu, L. Afanasiev, A. Arion, J. Dittrich, S. Manegold, N. Polyzotis, K. Schnaitter, P. Senellart, S. Zoupanos, and D. Shasha. The repeatability experiment of SIGMOD 2008. *SIGMOD Rec.*, 37:39–45, March 2008.