University of Washington Database Group

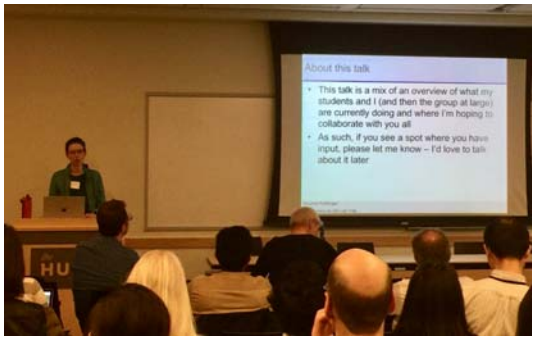
Newsletter - Winter 2018

We perform research in various aspects of data management systems, theory, and applications. Be sure to visit our website for news and upcoming events!

EVENTS

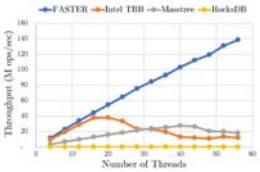
UWDB organized the **Northwest Database Society (NWDS)** annual meeting on UW campus in January where Professor Jeff Heer from UW gave the keynote talk on predictive interaction.

The meeting was a great gathering of over 100 data management researchers and students in the Pacific Northwest region, with participation from 10 different academic groups (all the way from Vancouver to Oregon!), and another dozen or so companies. The event provided a forum for researchers to gather and learn about ongoing projects from different groups. The event concluded with a poster session showing off 30 different projects. More info about the event on this website . We hope to see you at our annual meeting next year! Check out our website if you are interested in learning more about NWDS. Shown below (left to right) in these photos are distinguished speakers from the academia and industry: Rachel Pottinger (University of British Columbia), Maureen Stone (Tableau), Luna Dong (Amazon), Judy Cushing (Evergreen State College), Lin Li (Seattle University) and Vishaka Gupta (Intel).



I. RESEARCH HIGHLIGHTS

A. Recently Published and/or Accepted Papers



Title: FASTER: A Concurrent Key-Value Store with In-Place Updates

SIGMOD 2018

Authors: Badrish Chandramouli, Guna Prasaad, Donald Kossmann, Justin Levandosk, James Hunter. Mike Barnett .

Abstract:

Over the last decade, there has been a tremendous growth in data-intensive applications and services in the cloud. Data is created on a variety of edge sources, e.g., devices, browsers, and servers, and processed by applications in the cloud to gain insights or take decisions. Applications and services either work on collected data or monitor and process data in real time. These applications are typically update-intensive and involve a large amount of state beyond what can fit in main memory. However, they display significant temporal locality in their access pattern.

We developed FASTER, a persistent key-value store for state management. FASTER combines a highly cache-optimized concurrent hash index with a hybrid log: a concurrent log-structured record store that spans main memory and tiered storage while supporting fast in-place updates in memory. The hybrid log offers a self-tuning data organization capability to support a potentially drifting hot set, without requiring any fine-grained statistics or meta-data. FASTER extends the standard key-value store interface to handle read-modify-writes, blind and CRDT-based updates by leveraging dynamic code generation to provide native support for advanced user-defined update types. Experiments show that FASTER achieves orders-of-magnitude better throughput – up to 150M operations per second on a single machine – than alternative systems deployed widely today, and reaches bare-metal performance when the workload fits in memory.

This was part of Guna's Summer 2017 internship with Microsoft Research Database Group where he was mentored by Badrish Chandromouli.

Project website:
https://www.microsoft.com/en-us/research/project/faster/

Title: Automatically Leveraging MapReduce Frameworks for Data-Intensive Applications.

SIGMOD 2018

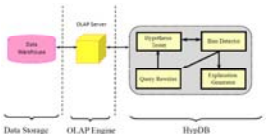
Authors: Maaz Bin Safeer Ahmad, Alvin Cheung

Abstract:

MapReduce is a popular programming paradigm for running large-scale data-intensive computation. Recently, many frameworks that implement that paradigm have been

developed. To leverage such frameworks, however, developers need to familiarize with each framework's API and rewrite their code. We present Casper, a new tool that automatically translates sequential Java programs to the MapReduce paradigm. Rather than building a compiler by tediously designing pattern-matching rules to identify code fragments to translate from the input, Casper translates the input program in two steps: first, Casper uses program synthesis to identify input code fragments and search for a functional specification of each fragment. The specification is expressed using a high-level intermediate language resembling the MapReduce paradigm. Next, each found specification is verified to be semantically equivalent to the original using a theorem prover. Casper then generates executable code from the specification, using either the Hadoop, Spark, or Flink API.We have evaluated Casper by automatically converting real-world sequential Java benchmarks to MapReduce. The resulting bench-marks perform up to 32.2x faster compared to the original, and are all translated without designing any pattern-matching rules

Project Website: http://casper.uwplse.org/



Title: Bias in OLAP Queries: Detection, Explanation, and Removal (Or Think Twice About Your AVG-Query).

SIGMOD 2018

Authors: Babak Salimi , Johannes Gehrke (Microsoft) and Dan Suciu (UW).

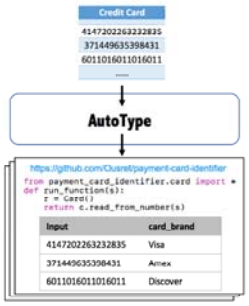
Abstract:

On line analytical processing (OLAP) is an essential element of decision-support systems. OLAP tools provide insights and understanding needed for improved decision making.

However, the answers to OLAP queries can be biased and lead to perplexing and incorrect insights. In this paper, we propose

HypDB, a system to detect, explain, and to resolve bias in decision-support queries. We give a simple definition of a biased query, which performs a set of independence tests on the data to detect bias. We propose a novel technique that gives explanations for bias, thus assisting an analyst in understanding what goes on. Additionally, we develop an automated method for rewriting a biased query into an unbiased query, which shows what the analyst intended to examine. In a thorough evaluation on several real datasets we show both the quality and the performance of our techniques, including the completely automatic discovery of the revolutionary insights from a famous 1973 discrimination case.

Project Website: http://db.cs.washington.edu/projects/hypdb/



Title: Synthesizing Type-Detection Logic for Rich Semantic Data Types using Open-source

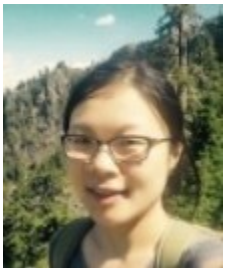
SIGMOD 2018

Authors: Cong Yan (University of Washington), Yeye He (Microsoft Research)

Abstract:

Given a table of data, existing systems can often detect basic atomic types (e.g., strings or numbers) for table columns. Recently, commercial data preparation and data analytics systems are starting to automatically recognize rich semantic types such as date-time, email address, etc., for such metadata can bring an array of benefits including better table understanding, precise data validation, and semantic data transformation. However, existing systems only detect a limited number of types using regular-expression-like patterns, which are often inaccurate, and cannot handle rich semantic types such as credit card and ISBN numbers that encode semantic validations (e.g., checksum).

We developed AutoType, a system that can synthesize type-detection functions for rich data types, by leveraging code from open-source repositories like GitHub. Users only need to provide a set of positive examples for a target data type and a search keyword, our system will automatically identify relevant code, and synthesize type-detection functions based on execution traces. We compiled a benchmark with 112 semantic types that are of interest for a commercial data preparation system. Our evaluation suggests that the proposed system can synthesize code to detect 84 such types at a high precision. Additionally, applying the synthesized type-detection code have resulted in a significant increase in the number of data types discovered compared to alternative approaches on web table columns.



Title: Columnstore and B+ tree -- Are Hybrid Physical Designs Important?

SIGMOD 2018 Industry Track:

Authors: Adam Dziedzic, Jingjing Wang, Sudipto Das, Bolin Ding, Vivek R. Narasayya, and Manoj Syamala.

Commercial Database Management Systems (DBMSs) support various physical design structures such as B+ tree indexes and columnar storage. It is well-understood that B+ trees suit transaction processing (OLTP) workloads while columnar storage serves analytical workloads more efficiently, but how to produce the optimal physical design for hybrid workloads remains unclear.

This paper results from Jingjing's internship at Microsoft back in 2014. In this project, together with her internship mentors at Microsoft Research, they extended the Database Tuning Advisor (DTA), a utility in Microsoft SQL Server that provides physical design recommendations, with the ability to automatically recommend a hybrid physical design for hybrid workloads. The hybrid physical design recommended by their approach can leverage the best of both B+ trees and columnstores, leading to orders of magnitude better execution costs compared to approaches that rely either on B+ tree-only or columnstore-only

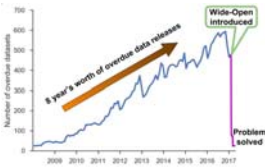
designs.

Please see related articles below:

https://blogs.technet.microsoft.com/dataplatforminsider/2017/01/10/announcing-columnstore-indexes-and-query-store-support-in-database-engine-tuning-advisor/

https://docs.microsoft.com/en-us/sql/relational-databases/performance/columnstore-index-recommendations-in-database-engine-tuning-advisor-dta

---



Title: Wide-Open: Accelerating Public Data Release by Automating Detection of Overdue Datasets.

PloS Biology 2017

Authors: Max Grechkin, Hoifung Poon , Bill Howe

Abstract:

Open data is a vital pillar of open science and a key enabler for reproducibility, data reuse, and novel discoveries. Enforcement of open-data policies, however, largely relies on manual efforts, which invariably lag behind the increasingly automated generation of biological data. To address this problem, we developed a general approach to automatically identify datasets overdue for public release by applying text mining to identify dataset references in published articles and parse query results from repositories to determine if the datasets remain private. We demonstrate the effectiveness of this approach on 2 popular National Center for Biotechnology Information (NCBI) repositories: Gene Expression Omnibus (GEO) and Sequence Read Archive (SRA). Our Wide-Open system identified a large number of overdue datasets, which spurred administrators to respond directly by releasing 400 datasets in one week.

This paper received coverage from from Nature News and The Scientist. Wide-Open is a new tool that may help advance open science by automatically detecting datasets that are overdue for publication.

Project website:
http://journals.plos.org/plosbiology/article?id=10.1371/journal.pbio.2002477

Title: How Not to Structure Your Database-backed Web Applications: A Study of Performance Bugs in the Wild.
ICSE 2018
Authors: Cong Yan, Alvin Cheung, Junwen Yang, Pranav Subramaniam, Shan Lu.

Abstract:
Many web applications use databases for persistent data storage, and using Object Relational Mapping (ORM) frameworks is a common way to develop such database-backed web applications. Unfortunately, developing efficient ORM applications is challenging, as the ORM framework hides the underlying database query generation and execution. This problem is becoming more severe as these applications need to process an increasingly large amount of persistent data. Recent research has targeted specific aspects of performance problems in ORM applications. However, there has not been any systematic study to identify common performance anti patterns in real-world such applications, how they affect resulting application performance, and remedies for them. In this paper, we try to answer these questions through a comprehensive study of 12 representative real-world ORM applications. We generalize 9 ORM performance anti-patterns from

more than 200 performance issues that we obtain by studying their bug-tracking systems and profiling their latest versions. To prove our point, we manually fix 64 performance issues in their latest versions and obtain a median speedup of 2× (and up to 39× max) with fewer than 5 lines of code change in most cases. Many of the issues we found have been confirmed by developers, and we have implemented ways to identify other code fragments with similar issues as well.

Cong Yan and collaborators from the University of Chicago worked on a tool for detecting web application performance issues. The tool, hyperloop, has been released! They performed a comprehensive study on existing open-source applications built with Ruby-on-Rails, and summarized common performance anti-patterns. These anti-patterns include common API misuse, loop invariant query motion, redundant queries, redundant data retrieval, inefficient webpage rendering, etc. A full list can be found in their ICSE paper. In this paper, they also propose fixes which are able to reduce page response time by 50% with only a few lines of code change. To help developers improve their applications, hyperloop was built to detect these anti-patterns by highlighting problematic code fragments.

Project website: https://hyperloop.cs.washington.edu

B. Short Papers, Demonstration Proposals, Workshops, and Technical Reports

Title: Cuttlefish: A Lightweight Primitive for Adaptive Query Processing
Authors: Tomer Kaftan, Magda Balazinska, Alvin

Cheung, Johanes Gehrke

Cuttlefish is a new primitive for adaptively processing online query plans by exploring operator implementations during query execution and exploiting the fastest ones using adaptive learning techniques. As modern data processing applications involve increasingly diverse operations beyond traditional relational algebra, designing traditional query optimizer rules and cost models to choose operator implementations becomes impractical. Instead, users and system developers can use Cuttlefish's simple yet flexible API in their applications to easily identify fast implementations for their workload setting, without having to pre-design any optimizations rules or cost models.

As the operators execute, Cuttlefish will choose and execute one of several candidate physical operator fragments on subsets of the input data at granularities supported by the operators. For example, a Cuttlefish may pick one algorithm per image for a convolution operator, and one join strategy per data partition for a distributed parallel join operator. Cuttlefish will balance exploration and exploitation to quickly identify the fastest physical operators.

Under-the-hood, Cuttlefish relies on statistcal learning techniques for solving the `Multi-armed Bandit Problem`. It supports using any available contextual features to automatically learn cost models specifying when different implementations should be used, although it does not require any contextual features. It is designed to work well in distributed settings, and it supports workloads where the best choice of operator varies over time and across machines.

We have prototyped Cuttlefish in Apache Spark, and used it to adaptively choose operators for image

convolution, regular expression matching, and relational joins. Our experiments have shown Cuttlefish-based adaptive convolution and regular expression operators can reach 72-99% of the throughput of an all-knowing oracle that always selects the optimal algorithm, even when individual physical operators are up to 105x slower than the optimal. Additionally, Cuttlefish can achieve join throughput improvements of up to 7.5x compared with Spark SQL's query optimizer.

Project
website: http://db.cs.washington.edu/projects/cuttlefish/

Title: Tighter Upper Bounds for Join Cardinality Estimates.

SIGMOD 2018 - Student Research Comptetion

Authors: Walter Cai; Advisors - Magda Balazinska and Dan Suciu

Abstract:

We develop a novel method to tighten join cardinality upper bounds. Our approach is as follows: leveraging data sketching, and randomized hashing we generate and tighten theoretical join cardinality upper bounds. We outline our base data structures and methodology, and how these bounds may be introduced to a traditional QO framework as a new statistic for physical join plan selection. We evaluate the tightness of our bounds on GooglePlus community graphs and successfully generate degree of magnitude upper bounds even in the presence of multiway cyclic joins. We also test the feasibility of using our bounds as cardinality point estimates inserted directly into the postgres QO and evaluate the results on the join order benchmark (IMDb) dataset.

Title: EZLearn: Exploiting Organic Supervision in Large-Scale Data Annotation.

NIPS LLD 2017 Workshop

Authors: Maxim Grechkin, Holfung Poon (Microsoft Research), Bill Howe.

Abstract:

Many real-world applications require large-scale data annotation, such as identifying tissue origins based on gene expression profiles and classifying images into semantic categories. Annotation classes are often numerous and subject to changes over time, and annotating examples has become the major bottleneck for supervised learning methods. In science and other high-value domains, large repositories of data samples are often available, together with two sources of organic supervision: a lexicon for the annotation classes, and text descriptions that accompany some data samples. Distant supervision has emerged as a promising paradigm for exploitingsuch indirect supervision by automatically annotating examples where the text description contains a class mention in the lexicon. However, due to linguistic variations and ambiguities, such training data is inherently noisy, which limits the accuracy in this approach. In this paper, we introduce an auxiliary natural language processing system for the text modality, and incorporate co-training to reduce noise and augment signal in distant supervision. Without any manually labeled data, our EZLearn system learned to accurately annotate data samples in functional genomics and scientific figure comprehension, even substantially outperforming state-of-the-art supervised methods trained on tens of thousands of annotated examples.

Project website: https://lld-workshop.github.io/papers/LLD_2017_paper_39.pdf

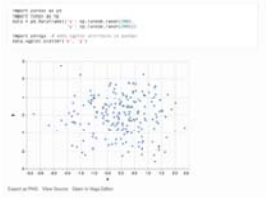
C. Projects Released

Brandon Haynes demonstrated LightDB, a database management system designed to efficiently manage virtual, augmented, and mixed reality (VAMR) video content. LightDB allows developers to declaratively express queries over VAMR video data and avoids the need to manually optimize workloads. Our prototype implementation offers up to 4x throughput improvements compared with existing systems. This work was also recently featured at the Northwest Database Society annual meeting , in the "Cross Cutting Systems Research" session at the UW affiliates research day, and at the industry affiliates research workshop. More details -including source code- are available on the project website.

Cosette is an automated prover for checking semantic equivalences of SQL queries that was introduced in July 2017. Cosette can be used to check the validity of query rewrites in database query optimizers, for grading students' data management homework, and for building a semantic caching layer of big data systems. We have started integrating Cosette with Apache Calcite, an open-sourced query optimization framework that powers many big data systems such as Hive and Drill. Cosette will help Calcite to improve software reliability by automatically finding bugs. The Cosette team is led by Shumo Chu and the team has recently visited Teradata and Microsoft SQL Server for potential technology transfer.

Dominik Moritz reports that the Interactive Data Lab released Vega-Lite 2.1 with support for geographic data, maps, and projections. Vega-Lite is a high-level

declarative language to create interactive data visualizations.



Jake VanderPlas has announced the release of PdVega , used to create interactive Vega-Lite plots in Python, using the familiar Pandas visualization API.

II. Recent Theses Defenses

Maxim Grechkin successfully defended his dissertation on algorithms and services for computational data curation. His work on jailbreaking unreleased datasets received coverage from The Scientist and Nature News, and his work on new machine learning methods that use noisy unstructured text metadata as a source of supervision recently won a best paper award at a NIPS 2017 workshop. That method, EZLearn, was used to create the current state-of-the-art corpus of curated gene expression data, beating even those datasets based on human-provided labels. And, his curated data is now being used by researchers at the UW working on cancer treatments. His most recent work aims to automatically validate certain kinds of scientific claims against public data, which simultaneously helps make data repositories more valuable while also holding scientists accountable for weakly reproducible results.

III. Internship and Hiring

Maaz Ahmad interned in the Creative Intelligence Lab at Adobe Research. During his internship, Maaz collaborated with Shoaib Kamil to develop automated techniques for rewriting legacy image processing code to newer architectures. Developers commonly use low-level hand optimizations to tune image processing pipelines to specific architectures (x86, CUDA etc). However, with the emergence of new architectures, these optimizations quickly become obsolete. Furthermore, such optimizations can make it difficult to re-target code to the newer architectures by obscuring program intent. During the internship, Maaz and Shoaib developed a prototype compiler that uses program synthesis and verification to lift low-level stencils written in C/C++ to Halide, a popular high-level domain specific language for image processing. The compiler is capable of automatically generating correct translations for stencil operations (blend, filter, smooth, color etc) found in the open-sourced OpenCV library. As an extension of this work, Maaz and Shoaib are collaborating with Alvin Cheung and Jonathan Ragan-Kelley to enable translation of entire image processing pipelines (for ex: Harris Corner Detection or Canny Edge Detector), thereby enabling greater optimizations.

Laurel Orr interned at Microsoft Research this past summer under the mentorship of Srikanth Kandula. She continued her research project from the previous summer, working on an approximate query processing technique that samples data blocks instead of data

rows to increase performance in big data queries without sacrificing too much accuracy.

Shrainik Jain was hired as a Software Engineering Intern at Snowflake Computing, wherein his work was on feature development for the SQL team. His projects included extensions to the SQL parser, new syntax support, prototyping a Materialized View and Clustering Advisor. A significant part of his internship involved research work on Materialized View selection. Furthermore, he continued his PhD research work on query workload analytics at Snowflake, as well.

III. FACULTY AWARD, NEW COURSES AND INVITED TALKS

A. Google Faculty Research Award

Magdalena Balazinska and **Alvin Cheung** together with collaborator **Prof. Luis Ceze r**eceived a **Google Faculty Research Award** for their project entitled "***Storage Management Techniques for Virtual Reality Video Data"*** in the context of their LightDB project

---

B. Invited Talks, New Courses, Workshops & Presentations

Magda Balazinska :

- Delivered the opening keynote talk at the ACM/IFIP/USENIX Middleware 2017 Conference

- Keynote talk at SOCC'17 where she talked about "Performance SLAs for Cloud Data Analytics" in September 2017.
- EECS Distinguished Speaker at Northwestern University. Feb 2018. In October 2017
- At University of Alberta: "The Myria Big Data Management System and Cloud Service" and the "The University of Washington eScience Institute" in October 2017.
- Invited talk at the @Scale Conference. Title: "Performance SLAs for Cloud Data Analytics". August 2017.
- Together with Prof. Jennifer Rexford (Princeton), David Culler (UC Berkeley), and Prof. Jeannette Wing (Columbia),
- Magda co-organized an NSF workshop on Enabling Computer and Information Science and Engineering Research and Education in the Cloud (Jan 8-9, 2018).
- Magda is co-organizing the first UW Data Science Summit , April 2018 sponsored by the eScience Institute.
- Developed a new course on "Scalable Data Systems and Algorithms" for our new Data Science Master's Program.

Alvin Cheung gave a talk at the StrangeLoop Developers' conference on leveraging domain-specific languages using the MetaLift framework. MetaLift uses program synthesis to automatically generate compilers that can transform programs written in general purposes (e.g., Java, C) into various domain-specific languages (e.g., Spark, Halide) in order to leverage the optimizations provided by the implementations.

Dan Suciu gave a keynote talk at the Northeast Database day at MIT on the communication cost in

parallel query processing. His survey paper on the same topic (with Paris Koutris and Semih Salihoglu) also appeared in Foundations and Trends in Databases, Dan Suciu also gave a distinguished lecture at Northeastern University, entitled "Rethinking Query Execution on Big Data", where he described novel worst-case optimal query processing algorithms in the presence of rich statistics on the input database.

Twitter Website Facebook

Computer Science & Engineering at the University of Washington is consistently ranked among the top programs in the nation. We educate tomorrow's innovators, conduct cutting-edge research in the principal areas of the field, lead a broad range of multi-disciplinary initiatives that demonstrate the transformative power of computer science and computer engineering, and are widely recognized for our success in promoting diversity.

The Paul Allen Center for Computer Science & Engineering

Copyright © 2018 UW Database Group, All rights reserved.

Our home page is:
http://db.cs.washington.edu/

Our people are:
http://db.cs.washington.edu/people.html

Want to change how you receive these emails?
You can update your preferences or unsubscribe from this list