# Introduction to Scientific Writing
# 01 Structure of Scientific Papers

SCIENCE
PASSION
TECHNOLOGY

**Matthias Boehm**

Graz University of Technology, Austria
Computer Science and Biomedical Engineering
Institute of Interactive Systems and Data Science
BMK endowed chair for Data Management

Last update: Oct 29, 2020

# Announcements/Org

- **#1 Virtual Lectures**
  - https://tugraz.webex.com/meet/m.boehm
  - Optional attendance (independent of COVID)
- **#2 Course Registrations** (as of Oct 28)
  - Changes in WS20/21
  - **Introduction to Scientific Writing**

**ISDS Group Boehm**

**20**

# Agenda

- **Data Management Group**
- **Course Organization, Outline, and Projects**
- **Structure of Scientific Papers**

# Data Management Group

https://damslab.github.io/

# About Me

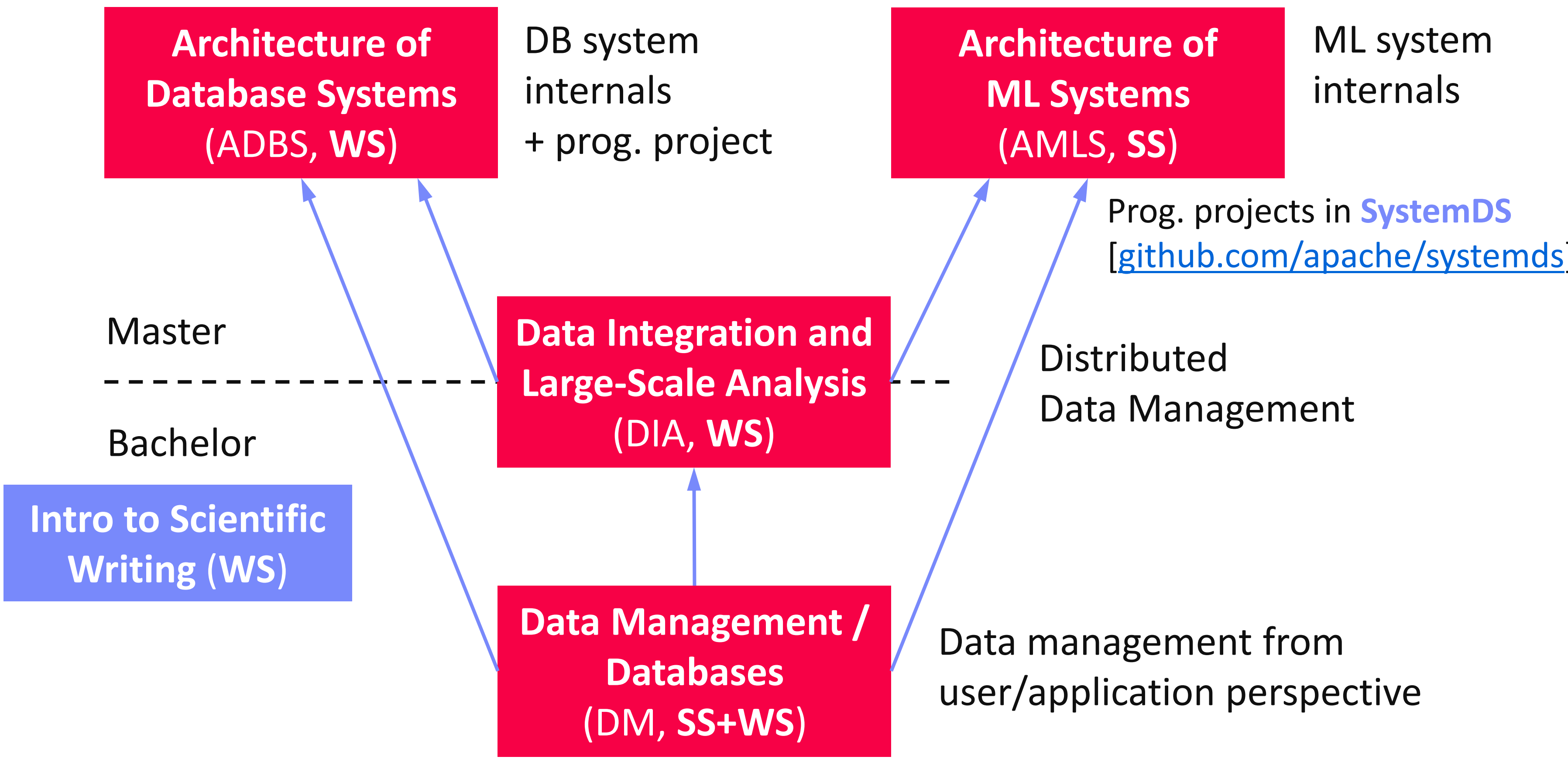
- **09/2018 TU Graz**, Austria
  - BMK endowed chair for data management
  - **Data management for data science**
    (ML systems internals, end-to-end data science lifecycle)

https://github.com/apache/systemds

- **2012-2018 IBM Research – Almaden**, USA
  - Declarative large-scale machine learning
  - Optimizer and runtime of **Apache SystemML**
- **2011 PhD TU Dresden**, Germany
  - Cost-based optimization of integration flows
  - Systems support for time series forecasting
  - In-memory indexing and query processing

DB group

# Data Management Courses



# Course Organization, Outline, Goals, and Projects

# Course Logistics


- **Staff**
  - Lecturer: Univ.-Prof. Dr.-Ing. Matthias Boehm, ISDS
  - Assistant: M.Sc. Shafaq Siddiqi, ISDS
- **Language**
  - Lectures and slides: **English**
  - Communication and examination: **English**/**German**
  - Submitted paper and talk: **English**
  - **Informal language** (first name is fine), immediate feedback
- **Course Format**
  - SE 1, **2 ECTS** (0.5 ECTS lectures + 1.5 ECTS paper/talk), bachelor
  - 3 lectures, **optional attendance**
  - Mandatory paper and presentation

# Course Logistics, cont.

- **Website**
  - https://mboehm7.github.io/teaching/ws2021_isw/index.htm
  - All course material (lecture slides) and dates
- **Video Recording Lectures (TUbe)? No**
  - Lectures and discussions via Webex
  - No recording in order to foster discussion, private presentations
- **Goals**
  - Understanding of / communication through scientific writing
  - Best practices for effective scientific reading, writing, and reproducibility
- **Grading**
  - **Overall:** pass/**fail** (no detailed 1-5 grades)
  - Includes submitted paper and final presentation
    (pass := adhere to given constraints + acceptable quality)

# Outline Lectures

- **01 Structure of Scientific Papers** [Oct 29, 6pm, **optional**]
- **02 Scientific Reading and Writing** [Nov 05, 6pm, **optional**]
- **03 Experiments, Reproducibility, and Projects** [Nov 12, 6pm, **optional**]

**...**

- **04 Project Presentations** [Jan 07, 6pm, **mandatory**]

# Paper Projects

**Alternative:** LV combined with bachelor thesis

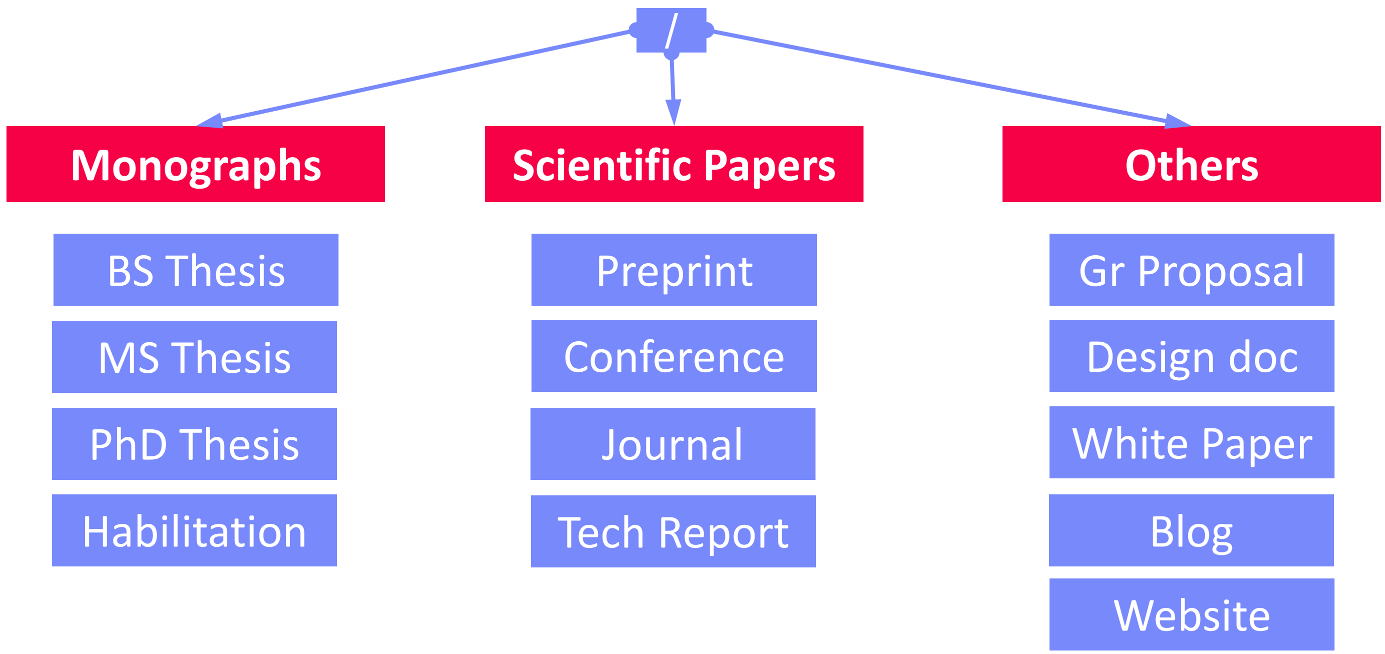
- **Team**
  - 1-4 person teams (w/ clearly separated responsibilities)
- **Project**
  - Pick from a given list of papers / groups of papers
  - **#1** Write short summary paper (#pages = 2 * team-size, written in LaTeX, ACM acmart template, document-class sigconf, PDF)
  - **#2** Prepare and present talk on paper summary (7min + 3min Q&A)
- **Timeline**
  - **Nov 12:** List of projects proposals, feel free to bring your own
  - **Dec 23:** paper submission via email to m.boehm@tugraz.at (11.59pm)
  - **Jan 07:** Final project presentation (all students)

# Structure of Scientific Papers

In Computer Science (Data Management)

# Overview Types of Scientific Writing

- **Classification of Scientific/Technical Documents**
  - Formal vs informal writing, cumulative?, single vs multi-author
  - Archival vs non-archival publications



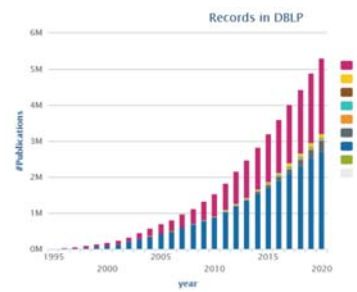

➔ **Scientific Writing Skills are crucial**
  - Different types of docs share many similarities

# Preparation

- **#1 Know your Audience**

- **#2 Get your Workflow in Order**
  - Writing: LaTeX (e.g., Overleaf, TeXnicCenter), versioning (e.g., git), templates
  - Plotting: R (plot, ggplot), Python (matplotlib), Gnuplot
  - Figures: MS Visio, Inkscape → pdf, eps, svg (vector graphics)

- **#3 Mindset: Quality over Quantity**
  - Aim for top-tier conferences/journals (act as filter)
  - Make the paper useful for others (ideas, evidence, code)
  - **Example** (my own theses/books)
    - Seminar (~bachelor), 5 months, **446** pages
    - Diploma (~master), 9 months, **274** pages
    - PhD thesis, 4 years, **237** pages
    - 1st book, 5+2 years, **157** pages




Your reader's time is a scarce resource

# Paper Writing and Publication Process

- **Research – Writing Cycle**
  - Read lots of papers
  - ~~Idea → Research → Writing → Document~~
  - Idea → Writing/Research → Document
  - Incremental refinement of drafts

[Recommended Reading]

[Simon Peyton Jones: How to write a great research paper, MSR Cambridge]


[Eamonn Keogh: How to do good research, get it published in SIGKDD and get it cited!, **KDD 2009**]


- **Paper Submission Cycle**
  - Blind vs double-blind submission
  - Revisions and Camera-ready
  - **Similar:** **bachelor**/master thesis
    → drafts to advisor / final version

Example SIGMOD 2020

**October 15, 2019**: Abstract submission
**October 22, 2019**: Paper submission
**December 10 - 11, 2019**: Author responses
**January 17, 2020**: Initial notification
**February 19, 2020**: Revised submission
**March 13, 2020**: Final notification
**April 12, 2020**: Camera ready due

- **Demo CMT**
  - Example SIGMOD 2021

# Paper/Thesis Structure by Example

- **Example Paper**

[Ahmed Elgohary, Matthias Boehm, Peter J. Haas, Frederick R. Reiss, Berthold Reinwald: Compressed Linear Algebra for Large-Scale Machine Learning. **PVLDB 2016**]

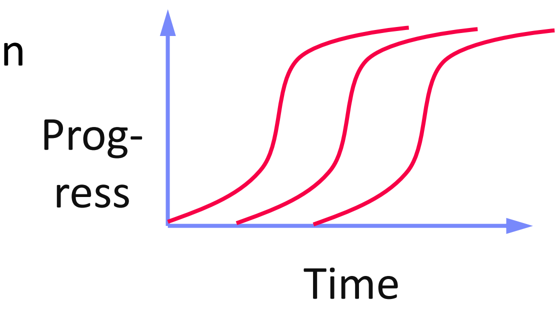
[Ahmed Elgohary, Matthias Boehm, Peter J. Haas, Frederick R. Reiss, Berthold Reinwald: Scaling Machine Learning via Compressed Linear Algebra. **SIGMOD Record 2017 46(1)**]

[Ahmed Elgohary, Matthias Boehm, Peter J. Haas, Frederick R. Reiss, Berthold Reinwald: Compressed Linear Algebra for Large-Scale Machine Learning. **VLDB Journal 2018 27(5)**]

[Ahmed Elgohary, Matthias Boehm, Peter J. Haas, Frederick R. Reiss, Berthold Reinwald: Compressed Linear Algebra for Large-Scale Machine Learning. **Commun. ACM 2019 62(5)**]

# Ideas and Topic Selection

- **Problem-Oriented Research**
  - Focus on problem/observation first, not your solution
  - **Discuss early ideas** with collaborators and friends
  - Develop your taste for good research topics
  - Topic selection needs time → **pipeline model**




- **Ex. Compressed Linear Algebra**
  - **Problem:** Iterative ML algorithms + memory-bandwidth-bound operations
    ➔ crucial to fit data in memory ➔ automatic lossless compression
  - **Sub-problems:** #rows>>#cols, column correlation, column characteristics
    ➔ column-wise compression w/ heterogeneous encoding formats

# Prototypes and Experiments

- **Worst Mistake: Schrödinger's Results**
  - Postpone implementation and experiments till last before the deadline
  - No feedback, no reaction time (experiments require many iterations)
  - **Karl Popper:** falsifiability of scientific results

- **Continuous Experiments**
  - Run experiments during survey / prototype building
  - Systematic experiments → observations and ideas for improvements
  - Don't be afraid of throw away prototypes that don't work

- **Ex. Compressed Linear Algebra**
  - Data characteristics inspired overall design of encoding schemes
  - Initially slow compression → dedicated sampling schemes and estimators
  - Initially slow compressed operations → cache-conscious operations, selected operations with better asymptotic behavior

# Title and Authors

- **List of Authors**
  - #1 by contribution (main, …, advisor)
  - #2 by last name

**Compressed Linear Algebra for Large-Scale Machine Learning**

Ahmed Elgohary[2]; Matthias Boehm[1], Peter J. Haas[1], Frederick R. Reiss[1], Berthold Reinwald[1]

[1] IBM Research – Almaden; San Jose, CA, USA
[2] University of Maryland; College Park, MD, USA

- **Title**
  - Descriptive yet concise
  - Short name if possible → easier to cite and discuss

**SPOOF: Sum-Product Optimization and Operator Fusion for Large-Scale Machine Learning**

Tarek Elgamal[2]; Shangyu Luo[3]; Matthias Boehm[1], Alexandre V. Evfimievski[1], Shirish Tatikonda[4]; Berthold Reinwald[1], Prithviraj Sen[1]

[1] IBM Research – Almaden; San Jose, CA, USA
[2] University of Illinois; Urbana-Champaign, IL, USA
[3] Rice University; Houston, TX, USA
[4] Target Corporation; Sunnyvale, CA, USA

**MNC: Structure-Exploiting Sparsity Estimation for Matrix Expressions**

Johanna Sommer, IBM Germany
Matthias Boehm, Graz University of Technology
Alexandre V. Evfimievski, IBM Research – Almaden
Berthold Reinwald, IBM Research – Almaden
Peter J. Haas, UMass Amherst

# Abstract

[Simon Peyton Jones: How to write a great research paper, MSR Cambridge]


**ABSTRACT**

Large-scale machine learning (ML) algorithms are often iterative, using repeated read-only data access and I/O-bound matrix-vector multiplications to converge to an optimal model. It is crucial for performance to fit the data into single-node or distributed main memory. General-purpose, heavy- and lightweight compression techniques struggle to achieve both good compression ratios and fast decompression speed to enable block-wise uncompressed operations. Hence, we initiate work on compressed linear algebra (CLA), in which lightweight database compression techniques are applied to matrices and then linear algebra operations such as matrix-vector multiplication are executed directly on the compressed representations. We contribute effective column compression schemes, cache-conscious operations, and an efficient sampling-based compression algorithm. Our experiments show that CLA achieves in-memory operations performance close to the uncompressed case and good compression ratios that allow us to fit larger datasets into available memory. We thereby obtain significant end-to-end performance improvements up to 26x or reduced memory requirements.

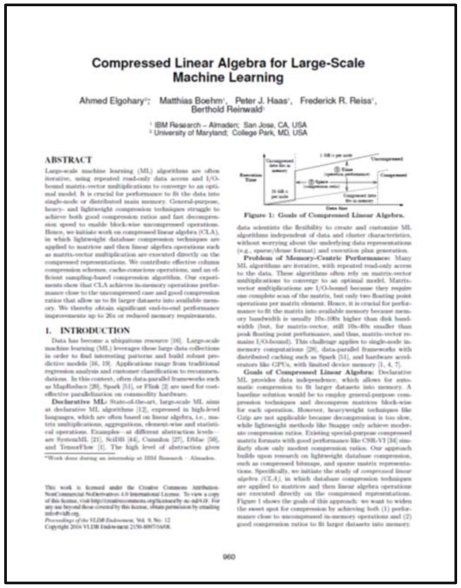
```
% 1. State the problem
Large-scale machine learning (ML) algorithms are often
iterative, using repeated read-only data access and I/O-bound
matrix-vector multiplications to converge to an optimal model.
It is crucial for performance to fit the data into single-node
or distributed main memory.
% 2. Say why it's an interesting problem
General-purpose, heavy- and lightweight compression techniques struggle to achieve
both good compression ratios and fast decompression speed to enable block-wise
uncompressed operations.
% 3. Say what your solution achieves
Hence, we initiate work on compressed linear algebra (CLA), in which lightweight
database compression techniques are applied to matrices and then linear algebra
operations such as matrix-vector multiplication are executed directly on the
compressed representations. We contribute effective column compression schemes,
cache-conscious operations, and an efficient sampling-based compression algorithm.
Our experiments show that CLA achieves in-memory operations performance close to the
uncompressed case and good compression ratios that allow us to fit larger datasets
into available memory.
% 4. Say what follows from your solution
We thereby obtain significant end-to-end performance improvements up to 26x or
reduced memory requirements.
```

# Introduction

- **Context** (1 paragraph)
- **Problems** (1-3 paragraphs)
- [Existing Work (1 paragraph)]
- [Idea (1 paragraph)]
- **Contributions** (1 paragraph)

**Contributions:** Our major contribution is to make a case for *compressed linear algebra*, where linear algebra operations are directly executed over compressed matrices. We leverage ideas from database compression techniques and sparse matrix representations. The novelty of our approach is a combination of both, leading towards a generalization of sparse matrix representations and operations. The structure of the paper reflects our detailed technical contributions:

- *Workload Characterization:* We provide the background and motivation for CLA in $\boxed{\text{Section 2}}$ by giving an overview of Apache SystemML, and describing typical linear algebra operations and data characteristics.
- *Compression Schemes:* We adapt several column-based compression schemes to numeric matrices in $\boxed{\text{Section 3}}$ and describe efficient, cache-conscious core linear algebra operations over compressed matrices.
- *Compression Planning:* In $\boxed{\text{Section 4,}}$ we further provide an efficient sampling-based algorithm for selecting a good compression plan, including techniques for compressed-size estimation and column grouping.
- *Experiments:* Finally, we integrated CLA into Apache SystemML. In $\boxed{\text{Section 5,}}$ we study a variety of full-fledged ML algorithms and real-world datasets in both single-node and distributed settings. We also compare CLA against alternative compression schemes.




### Introduction Matters

**Anchoring:** most reviewers reach their opinion after reading introduction and motivation and then look for evidence

[Eamonn Keogh: How to do good research, get it published in SIGKDD and get it cited!, **KDD 2009**]

# Writing the Paper (and more Experiments)

**02 Scientific Reading and Writing**
**03 Experiments and Reproducibility**

- **Easily Readable: Quality ∝ Time**
  - **Make it easy to skim the paper**
    → paragraph labels, self-explanatory figures (close to text), and structure
  - Avoid unnecessary formalism → as simple as possible
  - Shortening the text in favor of structure improves readability
  - Ex. Compressed Linear Algebra
    - Initial SIGMOD submission: **12+3 pages**
    - Final PVLDB submission: **12 pages**
      (+ more figures, experiments, etc)

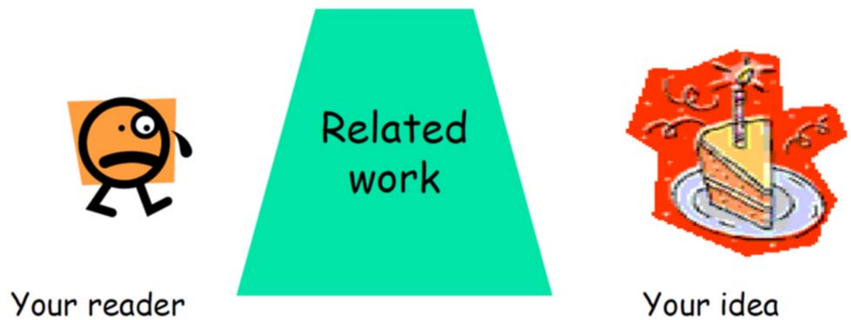
- **Solid, Reproducible Experiments**
  - Create, use, and share dedicated benchmarks / datasets
  - Avoid weak baselines, start early w/ baseline comparisons
  - Automate your experiments as much as possible
  - Keep repository of all scripts, results, and used parameters

# Related Work

- **Purpose of a “Related Work”-Section**
  - **Not** a mandatory task or to show you know the field
  - Put you work in context of related areas (~ 1 paragraph each)
  - Discuss closely related work
  - **Crisp separation from existing work** (what are the differences)

- **Placement**
  - Section 2 or **Section n-1**
  - Throughout the paper

[Simon Peyton Jones: How to write a great research paper, MSR Cambridge]


- **Give Credit**
  - Cite broadly, **give credit to inspiring ideas**, create connections
  - Honestly acknowledge **limitations of your approach**

# References

- **Setup**
  - Use LaTeX **\cite**{} and BibTeX
  - Use a consistent source of bibtex entries (e.g., DBLP)

```
inproceedings{StonebrakerBPR11,
  author    = {Michael {Stonebraker et al.}},
  title     = {{The Architecture of SciDB}},
  booktitle = {{SSDBM}},
  year      = {2011}
}
```

VLDB2016.bib

```
\bibliographystyle{abbrv}
\bibliography{VLDB2016}
```

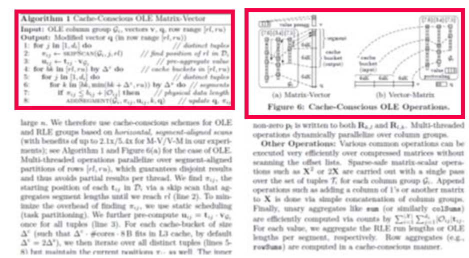

- **Different References Styles**
  - But, **not in footnotes** (unless required)
  - **8. REFERENCES**
    [1] M. Abadi et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. *CoRR*, 2016.
    [2] A. Alexandrov et al. The Stratosphere Platform for Big Data Analytics. *VLDB J.*, 23(6), 2014.
  - **References**
    [Al14] Alexandrov, A. et al.: The Stratosphere platform for big data J. 23/6, 2014.
    [AS14] Arap, O.; Swany, M.: Offloading MPI Parallel Prefix Scan the NetFPGA. CoRR abs/1408.4939/, 2014.
  - **References**
    Jaume Amores. 2013. Multiple instance classification: Review, taxonomy and comparative study. *Artificial Intelligence*.

# Dealing with Feedback / Criticism

- **Different Kinds of Feedback**
  - Casual discussion of early ideas
  - Comments on paper drafts
  - Reviewer comments (good and bad)

**# Always welcome feedback/criticism**
**# Address all feedback w/ sincere effort**

- **Example Compressed Linear Algebra**
  - SIGMOD Reviewer 2 (REJECT)
    - “The rewriting for q=Xv seems wrong: To compute q, one takes each row of the matrix X and multiplies it with the vector v.”
  - PVLDB Reviewer 3 (WEAK ACCEPT)
    - “I kinda disagree with the broad definition of declarative ML from the introduction.”



[Matthias Boehm et al.: Declarative Machine Learning - A Classification of Basic Properties and Types. **CoRR 2016**.]

- **Paper Rebuttal and/or Revision**
  - **Rebuttal**: seriously consider all feedback (in doubt agree), and answer with facts / ideas how to address the comments
  - **Revision** (conditional accept): address all revision requests

# Summary and Q&A

- **Data Management Group**
- **Course Organization, Outline, and Projects**
- **Structure of Scientific Papers**

- **Remaining Questions?**

- **Next Lectures**
  - 02 **Scientific Reading and Writing** [Nov 05]
  - 03 **Experiments, Reproducibility, and Projects** [Nov 12]