CS 8803-MDS

Human-in-the-loop Data Analytics

Lecture 10

09/26/22

Logistics

Kaushik's OH moved to Friday 10-11 this week.

Attending CoC career fair on Wednesday?

- Fill in Piazza poll
- If many people are attending, we might start the class a little later

Today's class

M4: A Visualization-Oriented Time Series Data Aggregation

Author: Shubham

Archaeologist: Bojun

Researcher: Jingfan

Practioner: Tanya

GENERAL INFORMATION

- Conference Overview
- Conference Officers
- PVLDB Review Board
- Industrial Track Committee

CALLS FOR CONTRIBUTIONS

- Papers
  - Research Track
  - Experiments and Analysis Track
  - Industrial Track
  - Innovative Systems Track
  - Vision Track
  - PhD Workshop
- Demonstrations
- Tutorials
- Panels
- Workshops
- Nominations

DATES AND GUIDELINES

- Important Dates

Overview

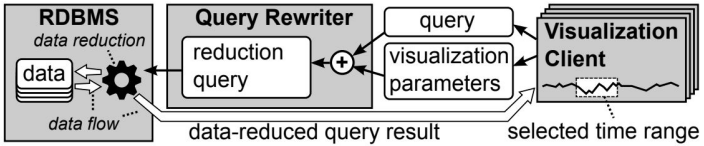
VLDB is a premier annual international forum for data management and database researchers, vendors, practitioners, application developers, and users. The conference will feature research talks, tutorials, demonstrations, and workshops. It will cover current issues in data management, database and information systems research. Data management and databases remain among the main technological cornerstones of emerging applications of the twenty-first century.

VLDB2014 will take place at Hangzhou, which is one of the best tourism cities in China. Hangzhou is also one of the eight ancient capitals in Chinese history and one in the first group of National Famous Historical and Cultural Cities. The West Lake in Hangzhou, known as the "earthly paradise", is one of the top attractions in China and abroad. Besides, Hangzhou is the hometown of tea and silk. Longjing Green Tea is the No. 1 among the top ten Chinese teas. attracts many tourists every year, and offers numerous opportunities for sightseeing (e.g., West Lake, Lingyin Temple, Qiantang River), outdoor activities (e.g., hiking, cycling, boating), cluture tasting (e.g., tea, silk, traditional cuisine, Grand Canal), as well as fun (e.g., Songcheng Theme Park, "Impression of West Lake" light opera).

M4: A Visualization-Oriented Time Series Data Aggregation (2014)

Bojun Yang

Summary



- Proposes time series aggregate dimensionality reduction strategy with a runtime O(n) without impairing resulting visualization
- C1: Query rewriting technique taking advantage of the fact that visualizations are inherently restricted to width x height pixels
  - Relies on relational operators and parameterized with width and height
- C2: Develop aggregation that only selects tuples with min and max, and first and last tuples having the min and max timestamp
- Results show M4 can reduce time user has to wait by 1 order of magnitude
  - Still provide correct tuples for high quality visualizations

Time Series Compression for Adaptive Chart Generation (2013)

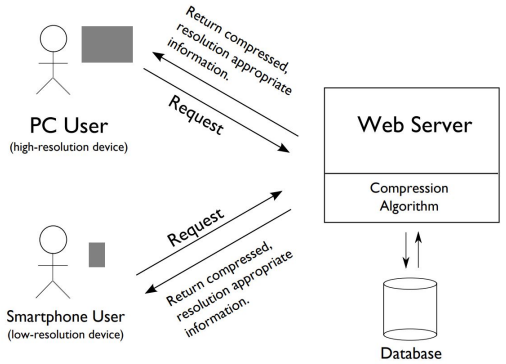
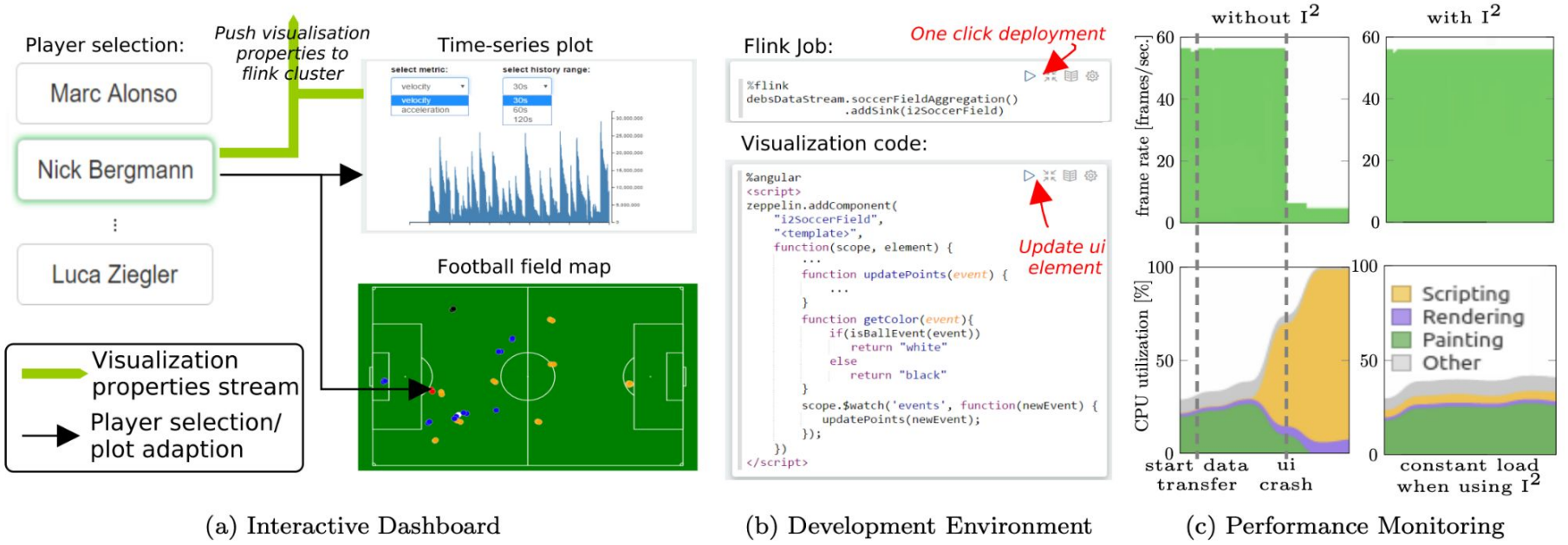


Fig. 2: High-Level Architecture Diagram

- Uses many compression techniques and 1 proposed implementation with knowledge of web client display sizes to transmit different amounts of data for different screen resolutions and sizes
- Algorithms used: No compression and run-length encoding (RLE), Visual aggregation (guarantees 1 point per pixel),Visual RLE (RLE but with range), Perceptually Important Points (PIPs), Piecewise Polynomial Approximation (proposed)
- 42% size-weighted avg savings
- Partially describes usage of visualization parameters for data reduction
  - Simplifies the problem: Does not consider visualization of the original time series resulting charts loose important detail
  - Applies the reductions outside of database
  - Use average aggregate

I²: Interactive Real-Time Visualization for Streaming Data (2017)



(a) Interactive Dashboard (b) Development Environment (c) Performance Monitoring

Figure 5: Selected screenshots from the I² demonstration.

I²: Interactive Real-Time Visualization for Streaming Data (2017)

- Enables users to specify real-time analysis programs and alter them on the fly
  - Apache Flink, Apache Zeppelin, Runtime Adaptive Operators
- Computes M4's 4 values per pixel column in a parallel data flow program for real-time visualization of incoming streaming data
  - Instead of SQL queries, they need parallelizable processing pipelines
- Techniques
  - Watermarks: keep track of smallest timestamp that is still covered by the live plot
  - Windowing: time window function to split stream into finite data chunks spanning the time of 1 pixel column. Calculate **M4** aggregates over these windows
  - Value Compression: map results of aggregation to value space of the y-axis

CS8803
M4: A Visualization-Oriented Time Series Data Aggregation

Researcher role: Jingfan Meng

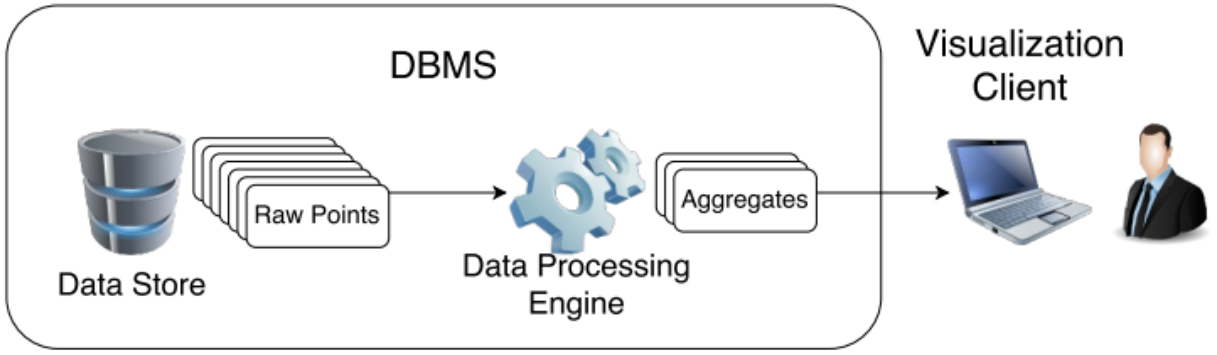
9/26/2022

Georgia Tech

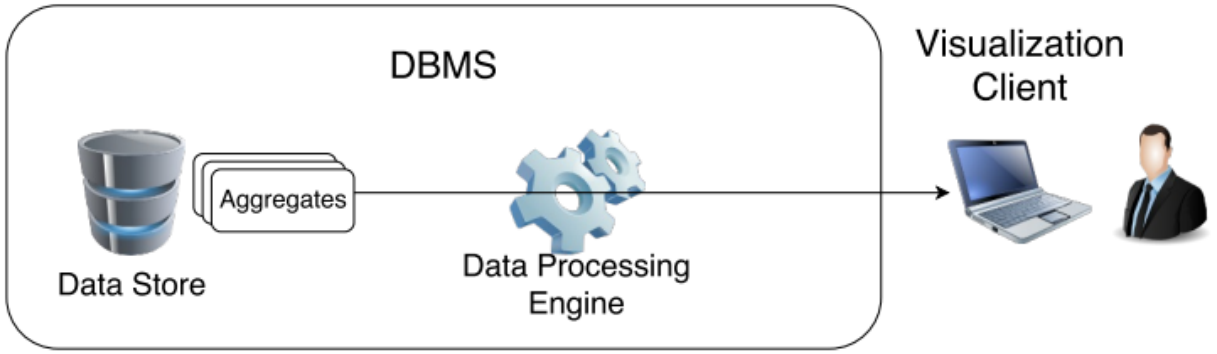
M4: Open problems

- M4 reduces the number of data transmitted on network by rewriting the original visualization query.
- It successfully reduces the time to transmit/load data, but does not reduce the time for the DBMS backend to answer the rewritten query.
- An exciting open problem is how to combine M4 with other approaches that speed up query answering.

M4 + AggPre = Mr. Plotter



(b) A design used by visualization tools like M4 and ScalaR: visualization client requests and renders data aggregates, which are computed by the database system on the fly



(c) Mr. Plotter's design: database system accelerates queries using precomputation; visualization client requests and renders precomputed data aggregates

- Mr Plotter: 2106.12505.pdf (arxiv.org)

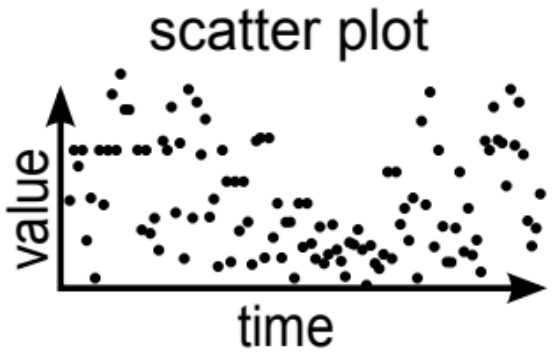
Georgia Tech

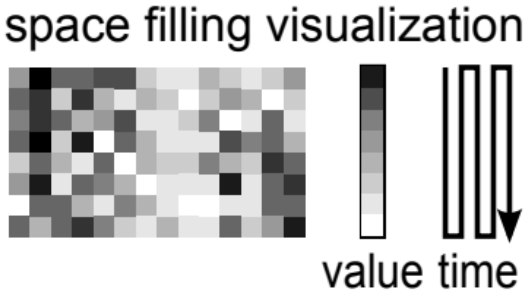
M4 + AggPre = Mr. Plotter

- To support user interaction (say zooming in by 1.2x), Mr. Plotter uses a hierarchy of precomputed aggregates so that each pixel column can be approximated by some aggregates.
- Limitation:
  - Reconstruction is not perfect since aggregate are not perfectly aligned with pixel columns.
  - The work only considers line charts.

M4 + (AQP++) = ?

- Let us look at the other two types of visualizations.
- Scatter plot displays nonempty groups by each pair of (time, value) that represents a pixel.
- Space filling visualization displays an aggregate value grouped by each time interval (pixel).
- We **propose** to speed up these group-by queries with AQP++.





Georgia Tech

Basic Idea

- To be simple, suppose we still have precomputed a power-of-two hierarchy of aggregates.
- For example, say a pixel column corresponds to time interval 0-98(s).
- We have two precomputed aggregates on 0-64 and 64-96 that covers most of this time interval.
- It remains to compute the aggregates online for the data in 96-98 only, and we can use only sampling to further speed up.

Basic Idea (cont.)

- When the user zooms in (say gradually), the previous and new pixel columns still have much overlaps.
- If stateful computation is allowed, we can store the aggregates on previous pixel columns, and use AQP++ to efficiently compute the difference.

Thanks!

Georgia Tech

M4: A Visualization-Oriented Time Series Data Aggregation

Practitioner Pitch

Tanya Garg

Summary of the Academic Paper

- M4 is a newly introduced, aggregation-based time series dimensionality reduction technique that provides error-free visualizations at high data reduction rates.
- Incorporates dimensionality reduction at the query level in a visualization driven query rewriting system and relies on relational operators and the raster space parameters.
- The technique groups a time series relation into w equidistant time spans, such that each group exactly corresponds to a pixel column in the visualization. For each group, M4 computes the aggregates min(v), max(v), min(t), and max(t) then joins the aggregated data with the original time series.
- By evaluating against different techniques using real-world datasets, its performance was critiqued.

Why should we use M4?

- Annual survey results: Need an inexpensive method to reduce the bandwidth consumption while querying time-series data for visualization.

Solution : Use dimensionality reduction techniques

- The current method gives inaccurate results => leads to incorrect inferences and subsequent revisions at times.

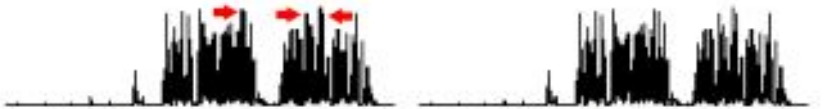
Solution and positive impact : M4 offers error-free visualizations, parameterized with the width of the raster space and gives a better time complexity of O(n)!

Can we implement it?

- Reuse current software implementation: Already have the query rewrite framework and compute 2 out of 4 values required for M4.

Current Method

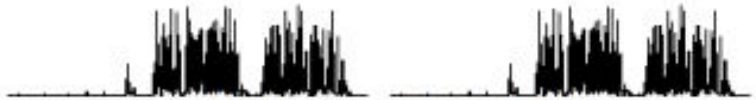
```
SELECT t,v FROM Q JOIN
(SELECT round($w*(t-$t1)/($t2-$t1)) as k, -- define key
        min(v) as v_min, max(v) as v_max  -- get min,max
        FROM Q GROUP BY k) as QA          -- group by k
ON k = round($w*(t-$t1)/($t2-$t1))        -- join on k
   AND (v = v_min OR v = v_max)           -- &(min|max)
```



b) resulting image c) expected image

Proposed M4

```
SELECT t,v FROM Q JOIN
(SELECT round($w*(t-$t1)/($t2-$t1)) as k, --define key
        min(v) as v_min, max(v) as v_max, --get min,max
        min(t) as t_min, max(t) as t_max  --get 1st,last
        FROM Q GROUP BY k) as QA          --group by k
ON k = round($w*(t-$t1)/($t2-$t1))        --join on k
   AND (v = v_min OR v = v_max OR         --&(min|max|
        t = t_min OR t = t_max)           --  1st|last)
```



b) resulting image == expected image

But should we use it?

- Works well with our current database that requires only line chart visualization.
- Do not need to revamp the current framework completely.
- **Negative Impact :**
  - **Cannot be used in the future if other forms of visualization such as scatter plots or space filling plots are required.**
  - **Engineers have to make sure that n_h = k*w to achieve the best results. Does not work well at low data volumes!**

Questions?

Contribution/Strengths

- M4’s approach relies only on relational operators for data reduction
- Apply data reduction on database layer, thus reducing data to be transferred and memory required to process the data.
- M4 method has a lower complexity of O(n) compared to line simplification techniques
- Most systems don't leverage visualization specifications, i.e., they do reduction irrespective of the visualization.
- Offers high data reduction rates (two orders of magnitude) with small time overhead and accuracy loss.

Limitation/Weaknesses

User perception

- The paper could be stronger if some surveys had been done on the perceived quality of visualizations made by different techniques. While M4 guarantees no error, it is difficult to infer how much this can benefit data analysts as they interact with the visualization tool.
- This form of pixel-by-pixel aggregation might still lead to lose in visualization semantics such as fluctuation among neighboring pixels and thus unable to represent the actual trends.

Limitation/Weaknesses

Performance

- … at all other points its performance is slightly poorer than the performance of MinMax and line simplification techniques
- If the visualization is not the final stage of work, any data processing would require the full fidelity data returned from the database, making the proposed method useless because reduced data are not fetched from the database.
- There is an assumption that there are no duplicate values per timestamp
- It is unclear whether the act of rewriting queries as min, max aggregates and waiting for the responses will enable an interactive user experience. For instance, the query execution time for M4 on the machine data (3600k rows) was 4 seconds

Limitation/Weaknesses

Scope

- Limited to line charts. There is no simple way to generate this method to error-free rasterization to scatter or space-filling plots.
- Authors do not talk about how the existing front-end tools handle screen resolution
- Grouping semantics that M4 uses does not allow for selection of non-aggregated values.
- Most of the discussions in the paper were made upon producing a quality and efficient two-color line visualization. It throws the reader off a little when the authors presented results on anti-aliased line visualization.

Questions

- Pre-aggregated data might not represent the raw data very well, cuz it have data loss, especially when considering high-volume time series data with a time resolution of a few milliseconds, and thus are subject to approximation errors. Why is M4 error free in this case?
- If M4 is based on pixels, does that mean using it on a 4k screen would actually make it slower?
- How to support editing?
- How to support zooming?

Discussion: AQP and visualization

How are these ideas similar/different between AQP and viz?

- Definition of error/accuracy
- Presentation of error
- Language (viz=SQL?)
- Use of precomputation techniques
- Use of sampling techniques

Next class

Data Science Tools

Benchmarking Spreadsheet Systems

Author: Abhinav, Harshal

Reviewer: Qiandong

Archaeologist: Cuong

Practioner: Ting