Toward Holistic Optimization of Data-Intensive Visualization Pipelines

Sebastian Breß, Volker Markl

Data Systems for Interactive Analysis
October 21, 2018

Technische Universität Berlin

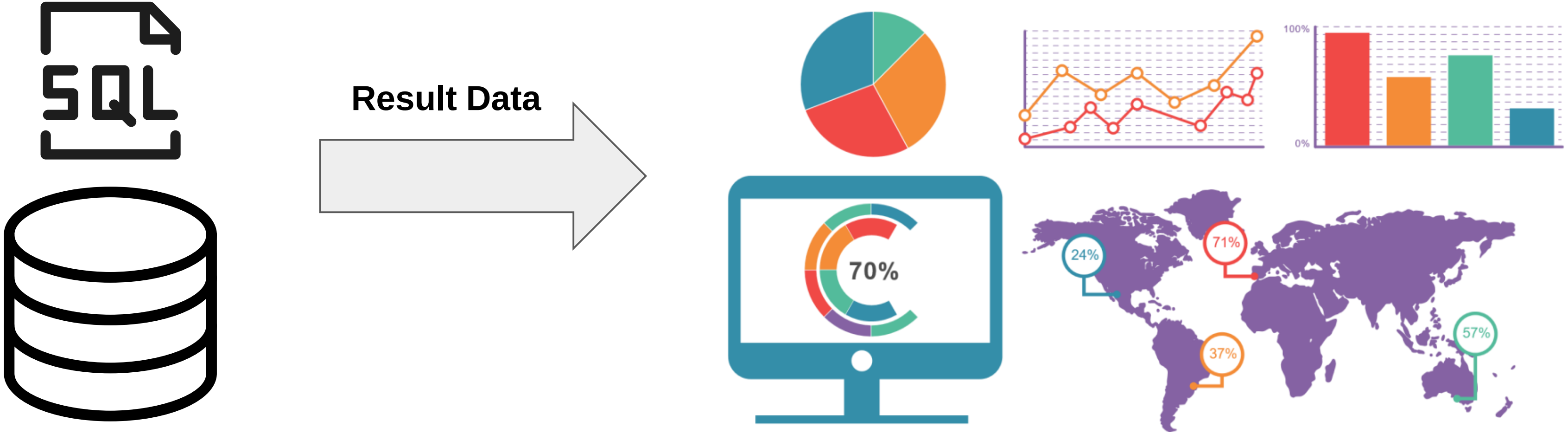
German Research Center for Artificial Intelligence GmbH

Data-Intensive Visualization Pipelines

Visualize large amounts of data interactively

Support efficient iterative, interactive data analysis by visualizations

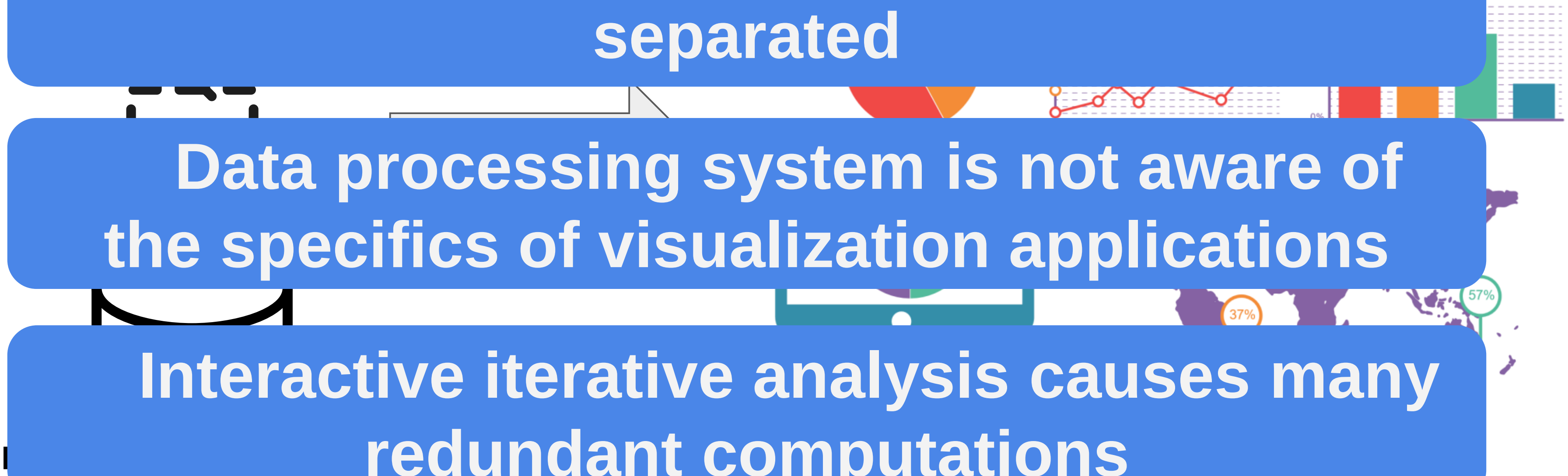
Data Analysis and Visualization



Data Processing System

Visualization System

Data Analysis and Visualization



- Data analysis and visualization are often separated
- Data processing system is not aware of the specifics of visualization applications
- Interactive iterative analysis causes many redundant computations

Challenges

Transfer Overhead: Ship large amounts of data to visualization system

Lost Potential: No holistic optimization between data processing and visualization

Interactivity: interactive visualizations and visualizing stream data are time critical

Integrating Data Processing and Visualization

Data Visualization Management System

Integrating Data Processing and Visualization

Data Visualization Management System

Integrating Data Processing and Visualization

No data transfer needed between systems

Processing exploits diverse properties of visualization to improve performance

Interactive iterative analysis tasks are responsive for large amounts of data

Data Visualization Management System

Optimizations for Visualization Pipelines

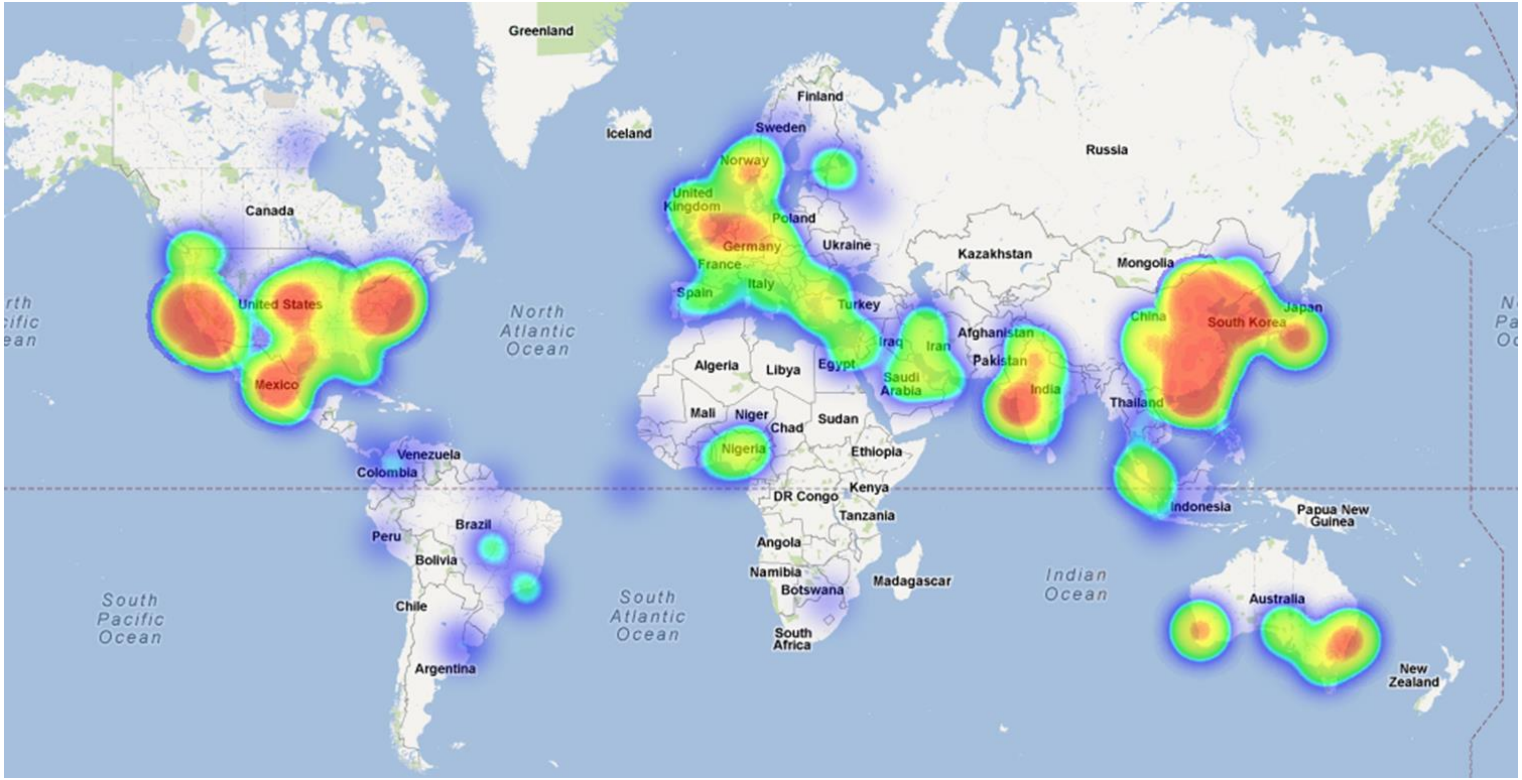
Optimizations in Visualization Pipelines

Visualization derived Filter Push-Down

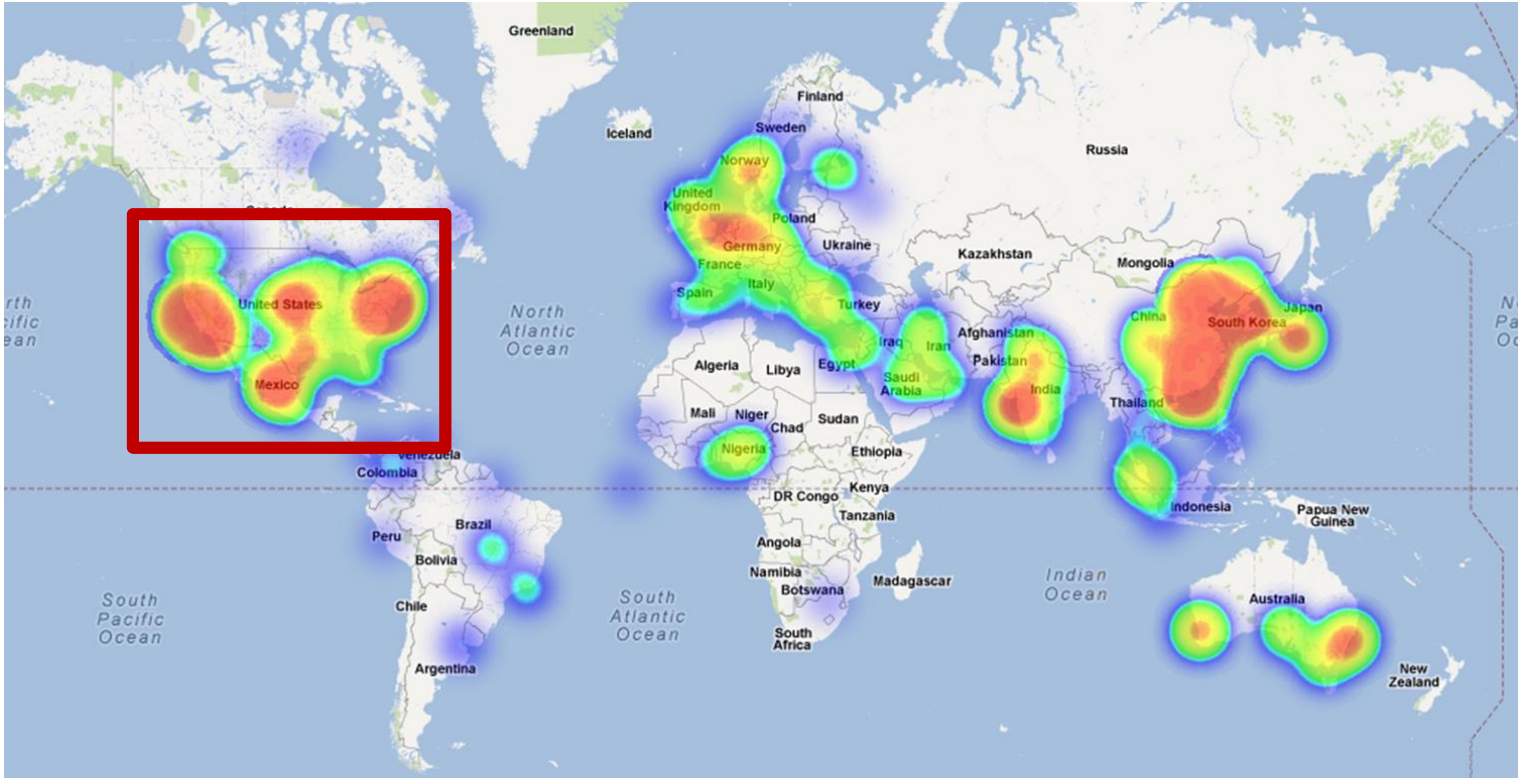
Pixel Perfect Compression

Transforming data processing problem into visualization problem

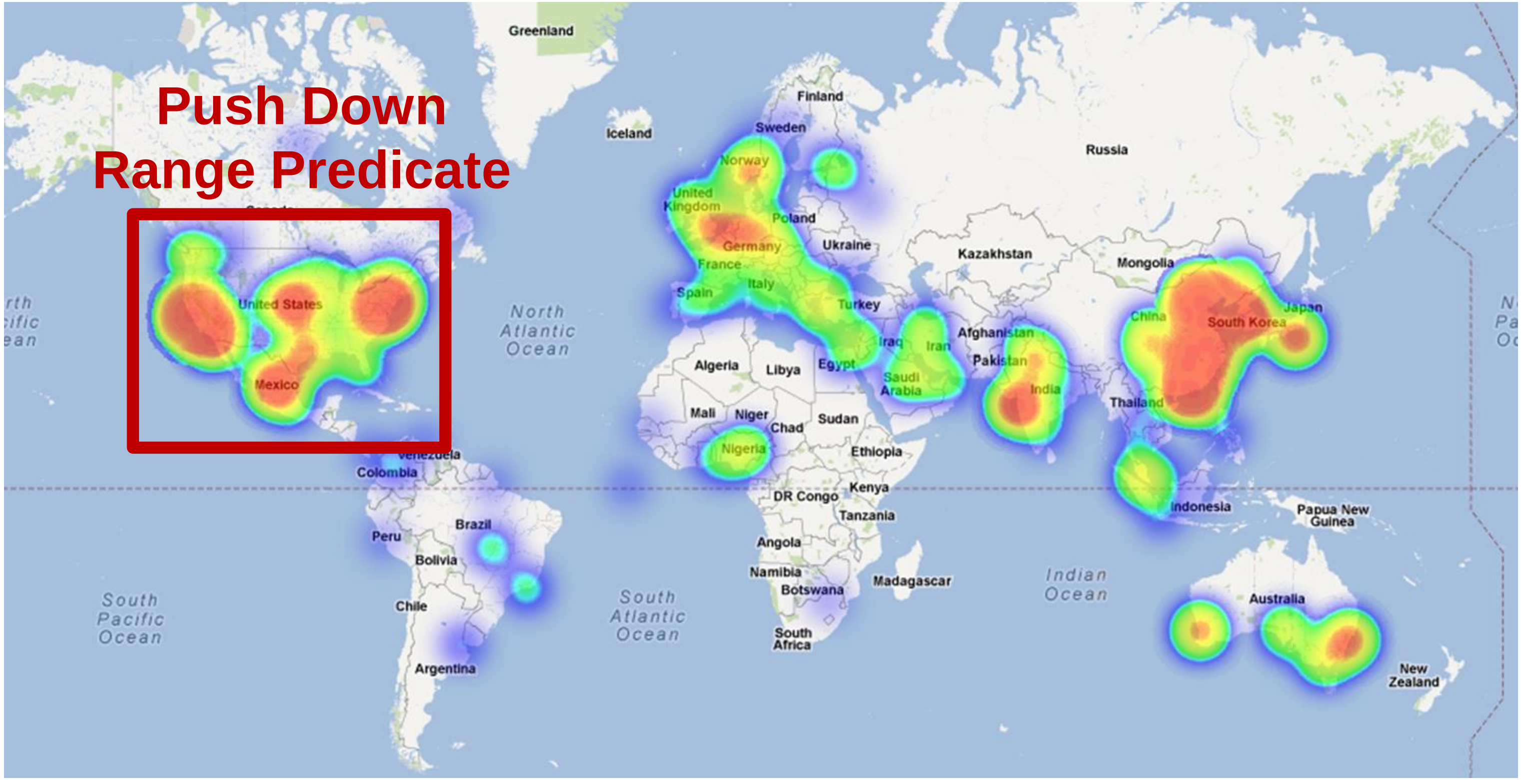
Visualization derived Filter Push-Down



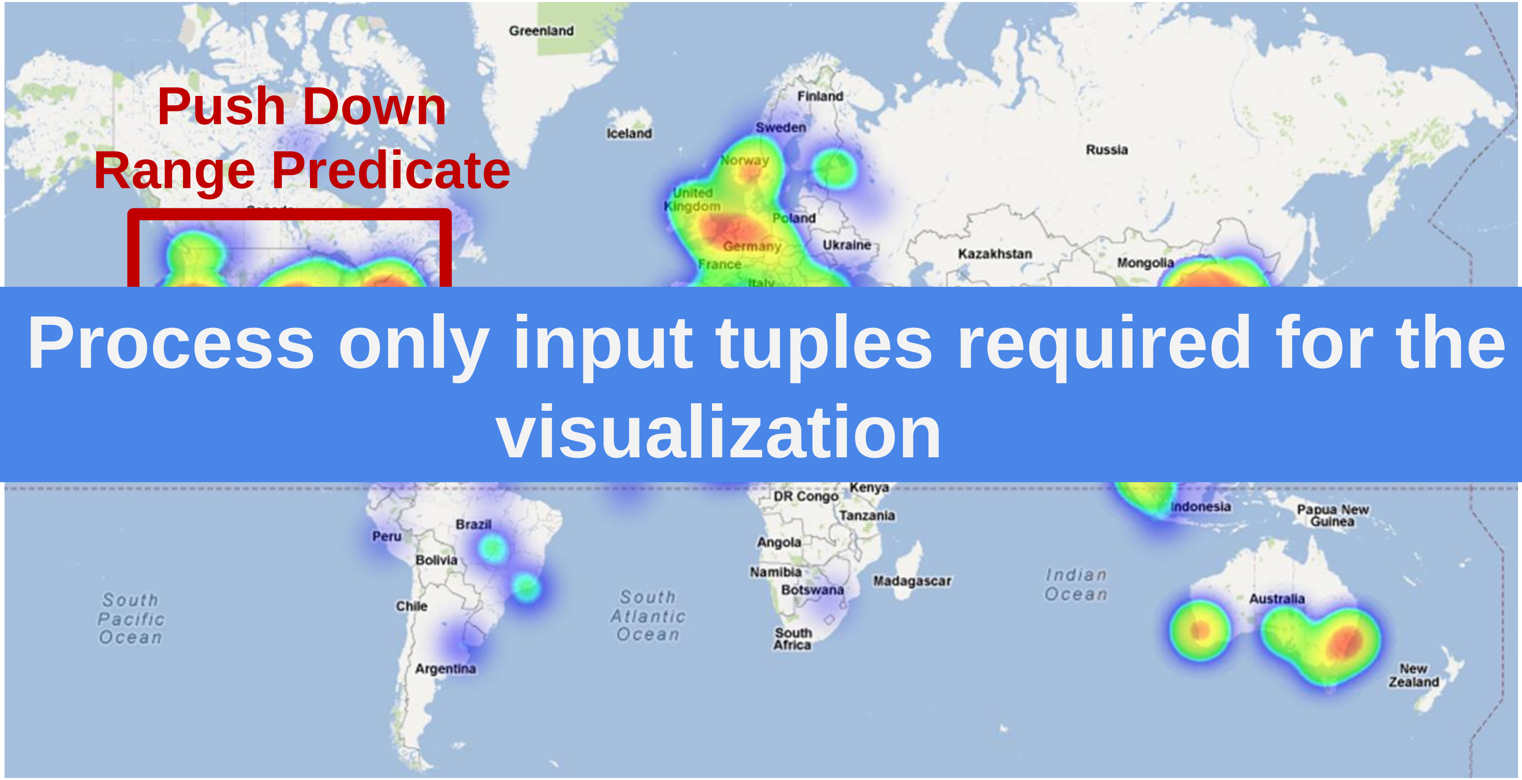
Visualization derived Filter Push-Down



Visualization derived Filter Push-Down



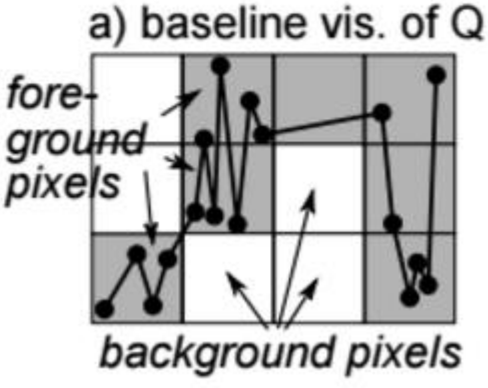
Visualization derived Filter Push-Down



Process only input tuples required for the visualization

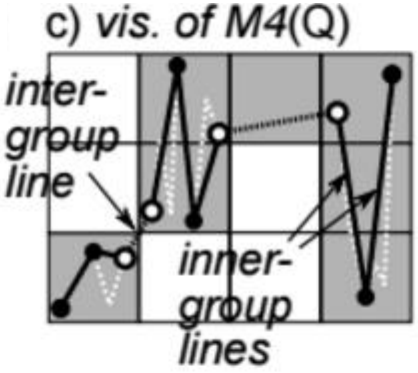
Pixel Perfect Compression

Big data time series often contain more data per pixel than we can display.



reduce data to
4 values per pixel column

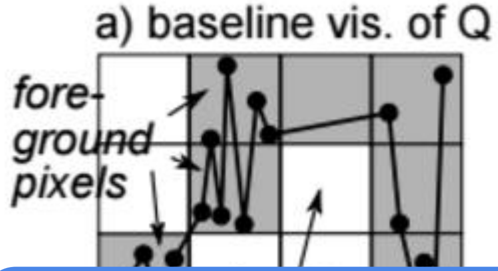
still provide a loss-free plot



Uwe Jugel, Zbigniew Jerzak, Gregor Hackenbroich, and Volker Markl. 2014. M4: a visualization-oriented time series data aggregation. Proc. VLDB Endow. 7, 10 (June 2014), 797-808.

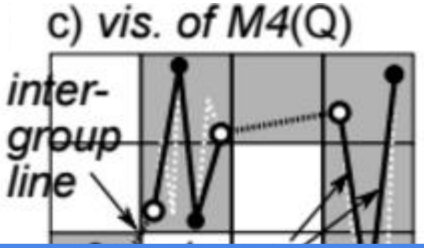
Pixel Perfect Compression

Big data time series often contain more data per pixel than we can display.



reduce data to
4 values per pixel column

still provide a loss-free plot



Compress query result before transferring it to visualization system

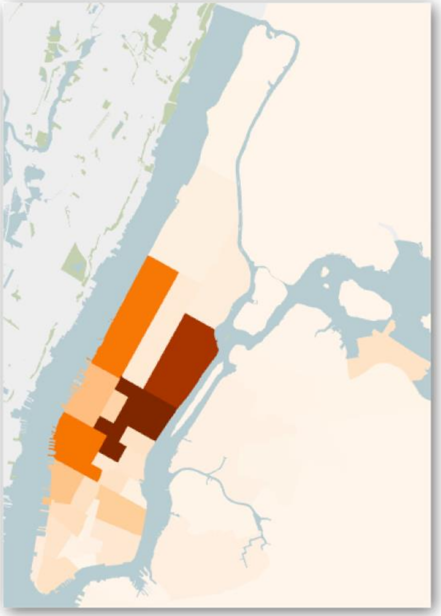
Uwe Jugel, Zbigniew Jerzak, Gregor Hackenbroich, and Volker Markl. 2014. M4: a visualization-oriented time series data aggregation. Proc. VLDB Endow. 7, 10 (June 2014), 797-808.

Rewrite Data Processing into Visualization Task

Spatial joins combine map data with other data (e.g., traffic)

Key Idea: Use image as an aggregation hash table

Optimization: Rewrite spatial join into rasterization task



Eleni Tzirita Zacharatou and others. GPU rasterization for real-time spatial aggregation over arbitrary polygons. Proc. VLDB Endow. 11, 3 (November 2017), 352-365.

Representing Visualization in Data Processing – A Systems Perspective

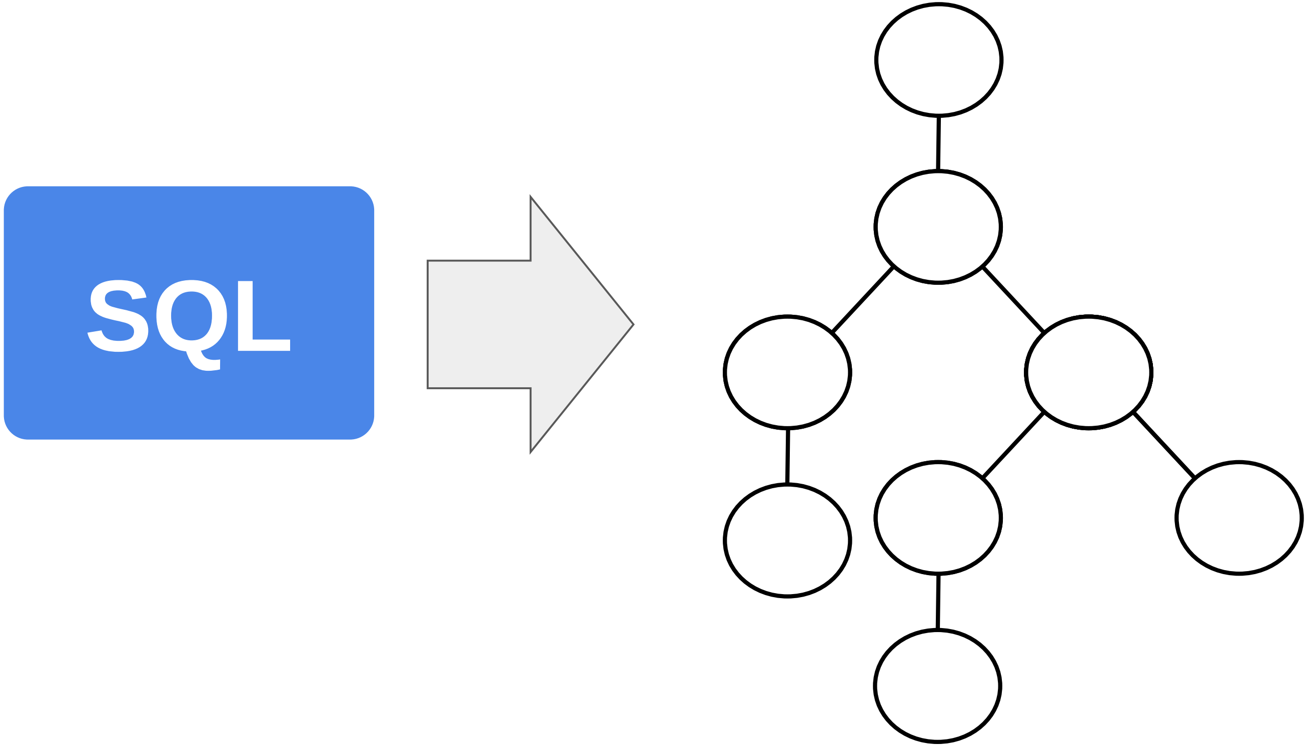
Represent Visualization in Data Processing

Introduce special visualization operators in data processing system

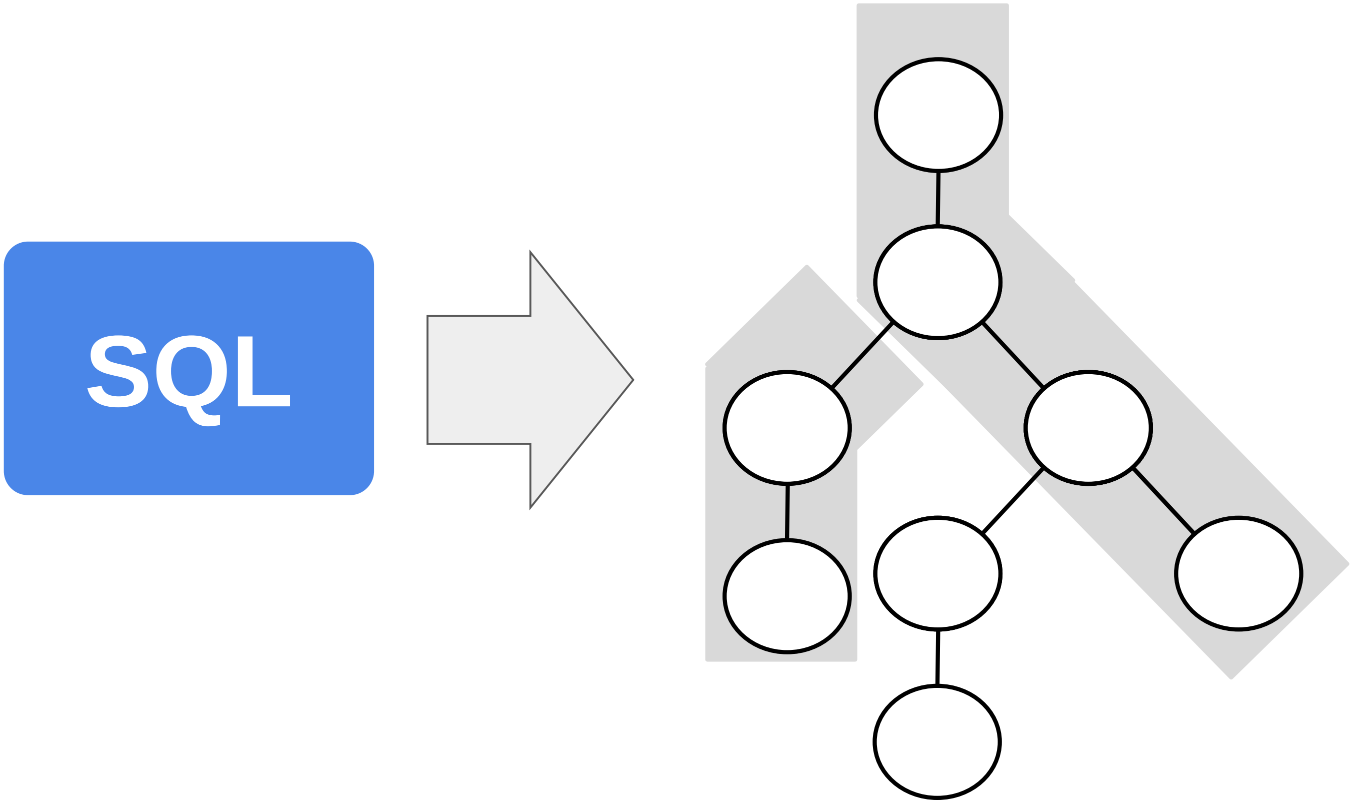
Optimizer needs to reason across relational and visualization operators

Fuse data processing and visualization operators for peak performance

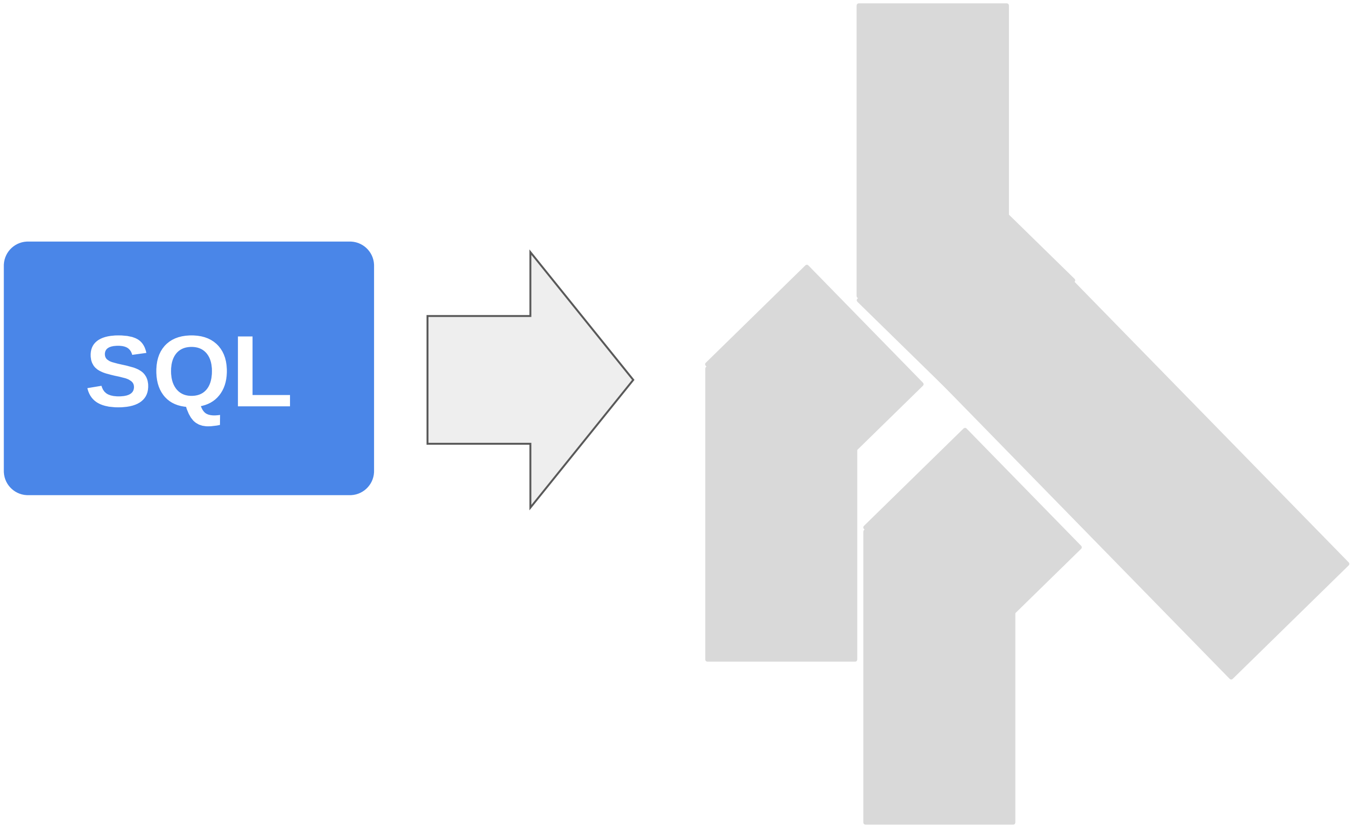
Operator Fusion by Query Compilation



Operator Fusion by Query Compilation



Operator Fusion by Query Compilation



Operator Fusion by Query Compilation


```
for(tuple t in table){
  if(age[id] < 25){
    /* more operators */
    OUTPUT(id[tid], age[tid]);
  }
 }
```

Query Compilation: Example

SQL Query

SELECT id, age FROM person WHERE age < 25;

Query Compilation: Example

SQL Query

SELECT id, age
FROM person
WHERE age < 25;

LOOP(person)

FILTER(age<25)

PROJECT(id, age)

Integrating Visualization in the Compilation

SQL Style
Visualization Query

Intermediate Program

DRAW HEATMAP
Lon, Lat, COUNT(age)
FROM person
WHERE age < 25
RANGE (x, y, w, h)

Integrating Visualization in the Compilation

SQL Style Visualization Query

DRAW HEATMAP
Lon, Lat, COUNT(age)
FROM person
WHERE age < 25
RANGE (x, y, w, h)

Intermediate Program

LOOP(person)

Integrating Visualization in the Compilation

SQL Style
Visualization Query

DRAW HEATMAP
Lon, Lat, COUNT(age)
FROM person
WHERE age < 25
RANGE (x, y, w, h)

Intermediate Program

LOOP(person)

FILTER(age<25)

Integrating Visualization in the Compilation

SQL Style Visualization Query

DRAW HEATMAP
Lon, Lat, COUNT(age)
FROM person
WHERE age < 25
RANGE (x, y, w, h)

Intermediate Program

LOOP(person)

FILTER(age<25)

HEATMAP (Lon, Lat, COUNT(age))

Integrating Visualization in the Compilation

SQL Style Visualization Query

DRAW HEATMAP
Lon, Lat, COUNT(age)
FROM person
WHERE age < 25
RANGE (x, y, w, h)

Intermediate Program

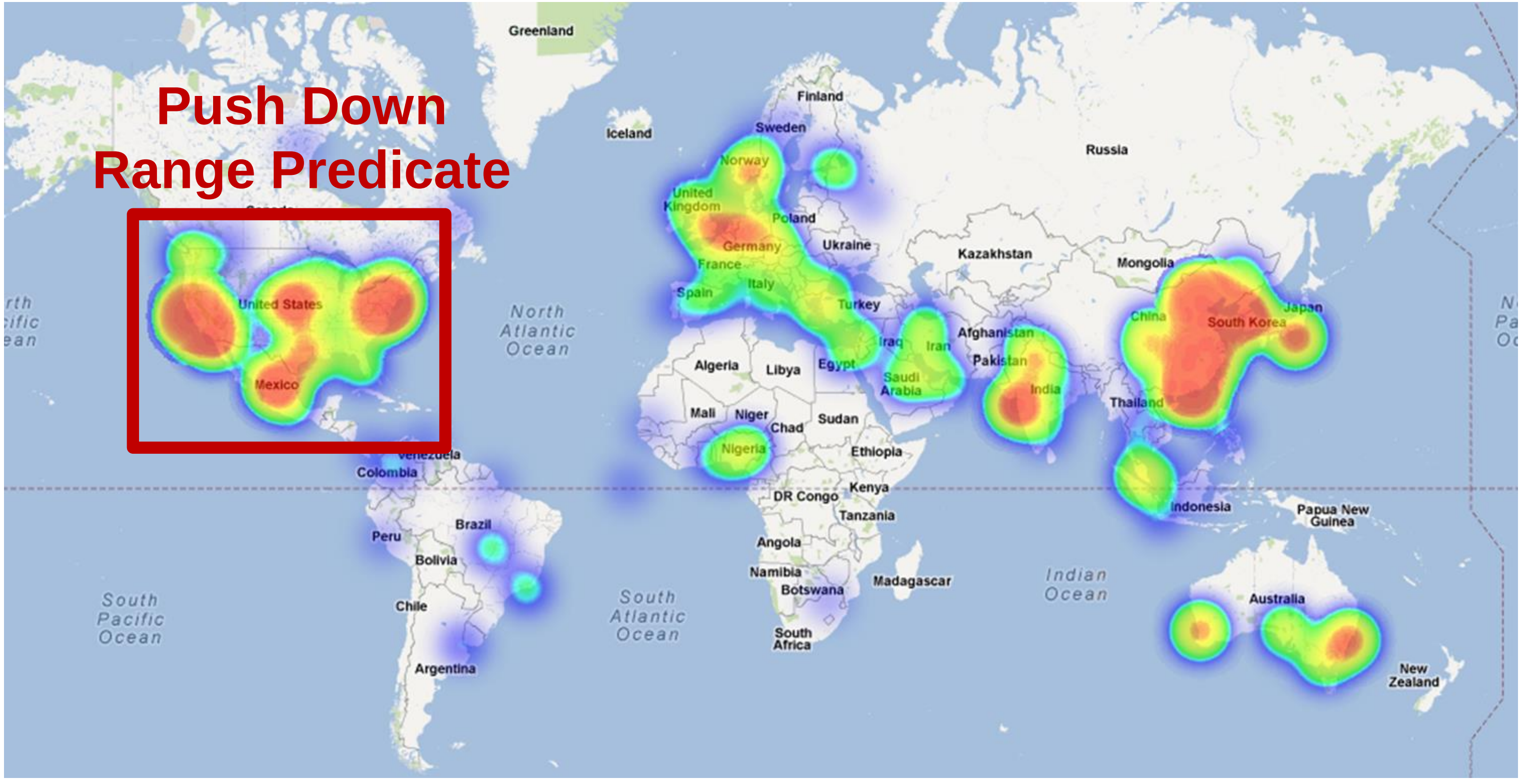
LOOP(person)

FILTER(age<25)

HEATMAP (Lon, Lat, COUNT(age))

ZOOM (x, y, w, h)

Recall the Filter Push Down Optimization



Optimize Based on Intermediate Representation

SQL Style Visualization Query

Intermediate Program

DRAW HEATMAP
Lon, Lat, COUNT(age)
FROM person
WHERE age < 25
RANGE (x, y, w, h)

Optimize Based on Intermediate Representation

SQL Style Visualization Query

DRAW HEATMAP
Lon, Lat, COUNT(age)
FROM person
WHERE age < 25
RANGE (x, y, w, h)

Intermediate Program

LOOP(person)

Optimize Based on Intermediate Representation

SQL Style Visualization Query

DRAW HEATMAP
Lon, Lat, COUNT(age)
FROM person
WHERE age < 25
RANGE (x, y, w, h)

Intermediate Program

LOOP(person)

FILTER(age<25)

Optimize Based on Intermediate Representation

SQL Style Visualization Query

DRAW HEATMAP
Lon, Lat, COUNT(age)
FROM person
WHERE age < 25
RANGE (x, y, w, h)

Intermediate Program

LOOP(person)

FILTER(age<25)

FILTER((Lon,Lat) in RANGE(…)(…))

Optimize Based on Intermediate Representation

SQL Style Visualization Query

DRAW HEATMAP
Lon, Lat, COUNT(age)
FROM person
WHERE age < 25
RANGE (x, y, w, h)

Intermediate Program

LOOP(person)

FILTER(age<25)

FILTER((Lon,Lat) in RANGE(…)(…))

HEATMAP (Lon, Lat, COUNT(age))

Take Home

Combining data processing and visualization enables new optimizations

An extensible intermediate representation captures semantics of both domains

Achieve peak performance by integrating visualization into query compilation