WebGPU for Scalable Client-Side Aggregate Visualization

Gerald Kimmersdorfer, Dominik Wolf, Manuela Waldner
TU Wien, Institute of Visual Computing & Human-Centered Technology, Austria

MOTIVATION

Modern web visualization frameworks scale poorly for huge datasets. Aggregate visualization limits the number of data points to be rendered, but bin recomputation becomes the bottleneck.

WebGPU provides an API for GPGPU in web browsers via compute shaders. This offers a **significant improvement in performance** through parallelization.

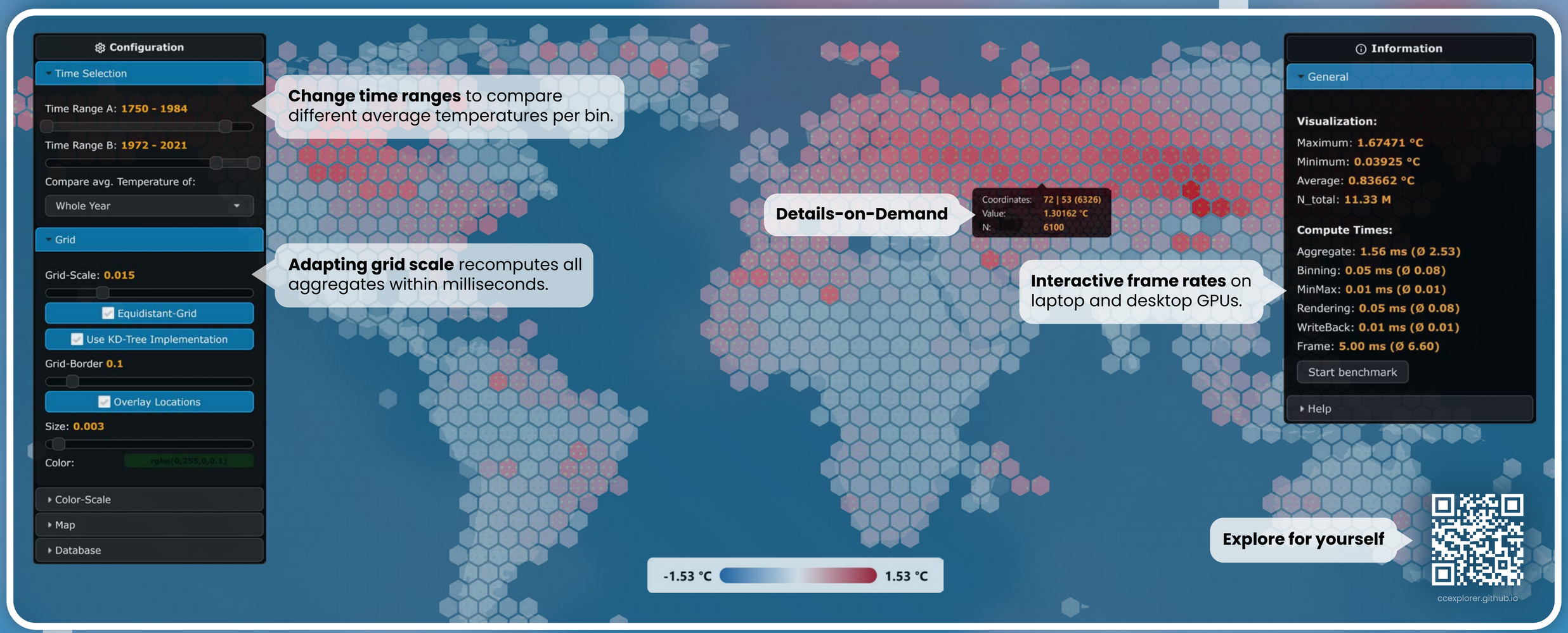
OUR GOAL

Showcase the potential of WebGPU for aggregate visualization.

Enable users to **filter and aggregate millions of data points in realtime** to explore temperature changes around the world.

Render aggregated data as hexagonal bins superimposed on a world map.

CLIMATE CHANGE EXPLORER



BEHIND THE SCENES

Pre-Processing - *Offline Step*
Compression of ~10M temperature entries at 5,165 different locations (Berkeley Earth Local Temperature Dataset).

<< **Recompute** if time range or grid scale changes <<

Aggregation - *WebGPU Compute Pipeline*
Average temperatures per location depending on specified time range.

Binning - *WebGPU Compute Pipeline*
Calculate average temperature per bin. Neighborhood search utilizing a *k*-d tree.

RESULTS

We can maintain **interactive frame rates** with millions of data points. WebGPU's compute and rendering pipelines can **significantly increase the performance of client-side aggregate visualization** compared to state-of-the-art solutions.

Rendering - *WebGPU Rendering Pipelines*
Classical rendering pipelines for the world map and the hexagon overlay.

Min-Max - *WebGPU Compute Pipeline*
Evaluate the minimum and maximum grid values for the color scale. Parallel reduction leads to GPU peak performance.

TU WIEN
FACULTY OF INFORMATICS
INSTITUTE OF VISUAL COMPUTING & HUMAN-CENTERED TECHNOLOGY
RESEARCH UNIT OF COMPUTER GRAPHICS

FWF Österreichischer Wissenschaftsfonds
P 36453