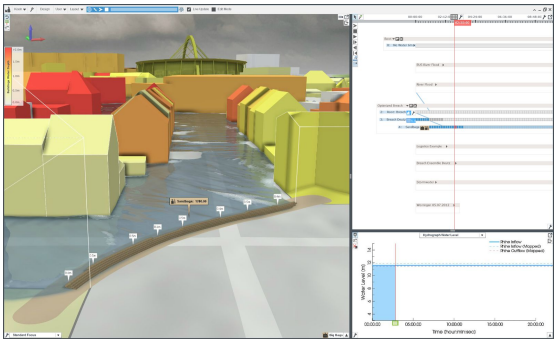
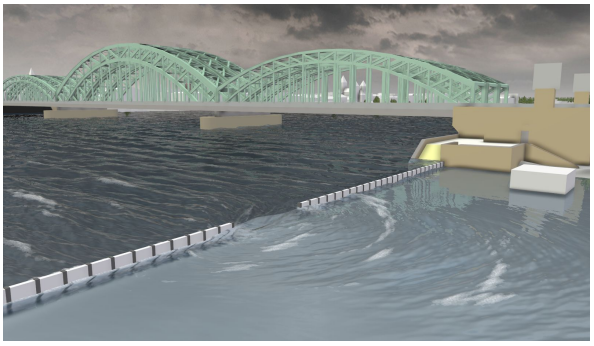
Student Projects in Visdom

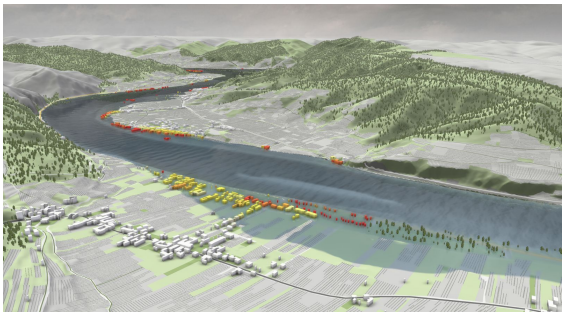
October, 2017

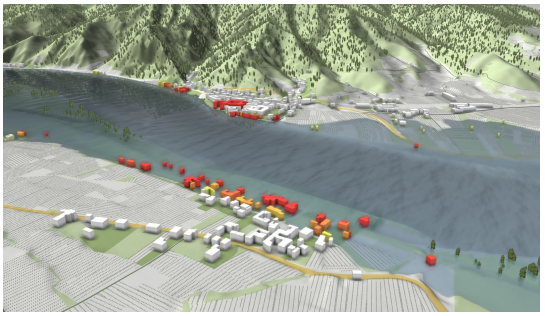
(http://visdom.at, mail: jwaser@vrvis.at)

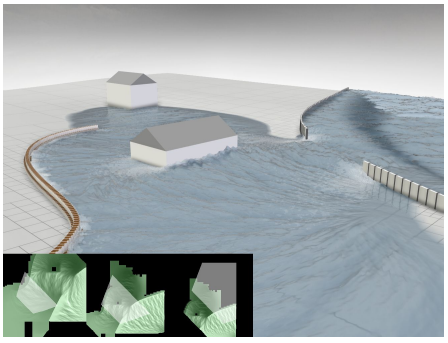
Main Goal of Visdom: Visual decision support based on simulation scenarios
Research Fields: Rendering, Simulation, Visualization, Visual Analytics
Main Application: Disaster management

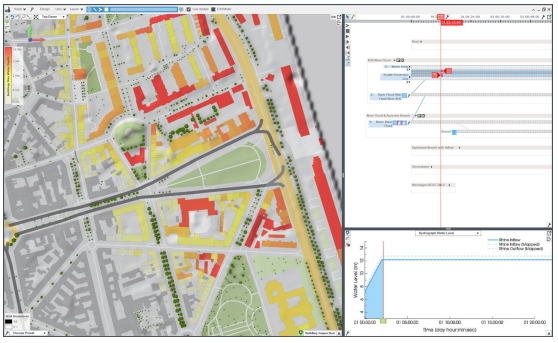








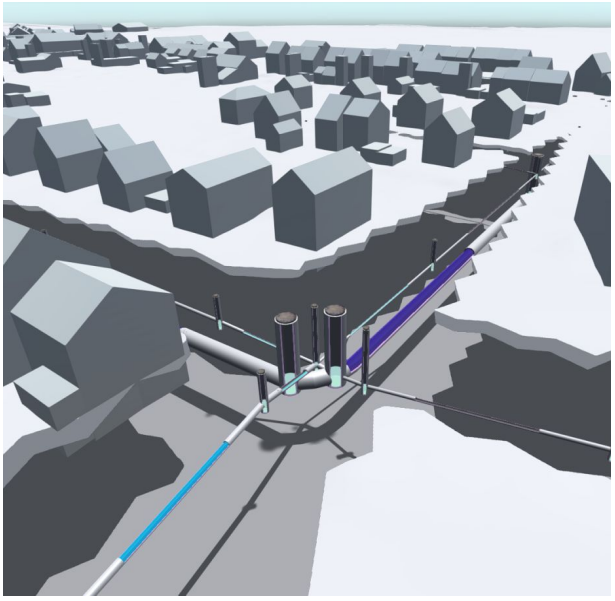




Topic: Impostor rendering for geospatial applications

Geometric primitives such as spheres, boxes, and cylinders can be rendered efficiently by rendering a 2D plane over the desired screen area (impostor) and then evaluating a function describing the geometric object in a fragment shader. Often, this is even more efficient than hardware instancing and allows to render thousands of simple objects. Possible applications in Visdom include

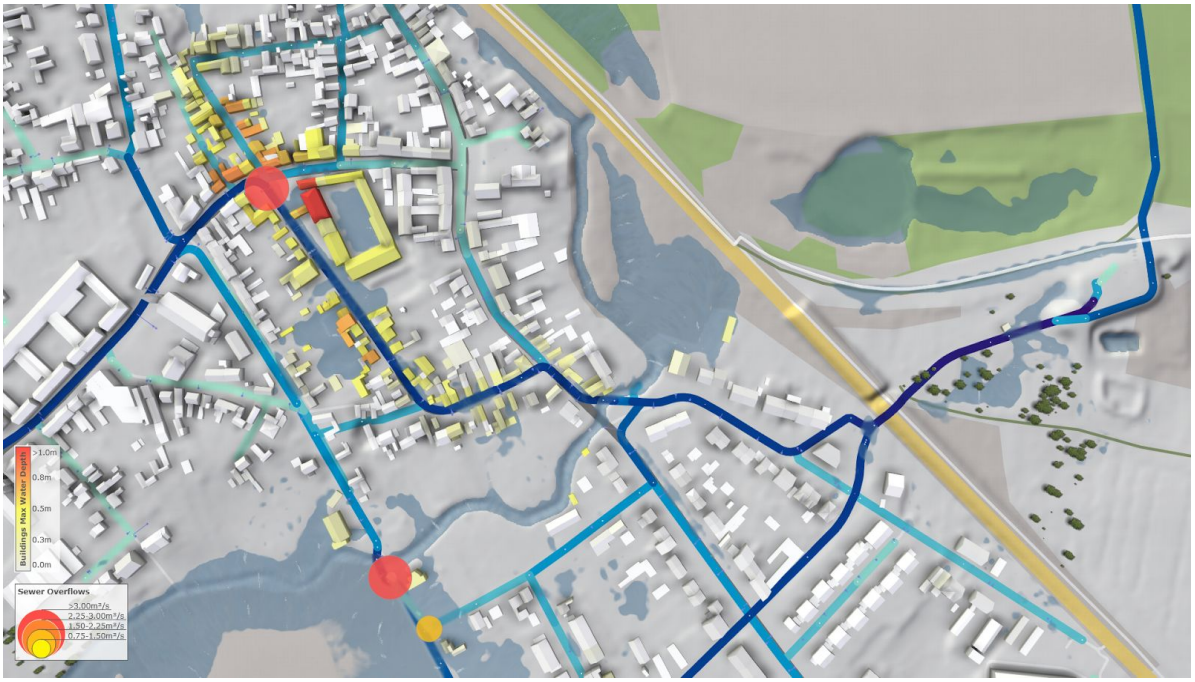
- Sewer network including special structures (pumps, weirs, …)
- Subways
- 3D lines in general
- Vegetation (e.g., tree types)
  https://www.offenedaten-koeln.de/dataset/baumkataster-koeln
  Visualization on vegetation, e.g, visualize effect of forest on retention; Infiltration, Roughness; Wildfire
- View-dependent, curved glyphs



Topic: Decal rendering

A lot of geospatial data given in 2D need to be visualized on the 3D terrain efficiently and without artifacts like z-fighting. These include roads land use information, building footprints, a city's sewer network, and feedback lines for user interaction. In Visdom, these data are rendered as decals with a two-pass algorithm using a stencil buffer. Extensions of this technique include

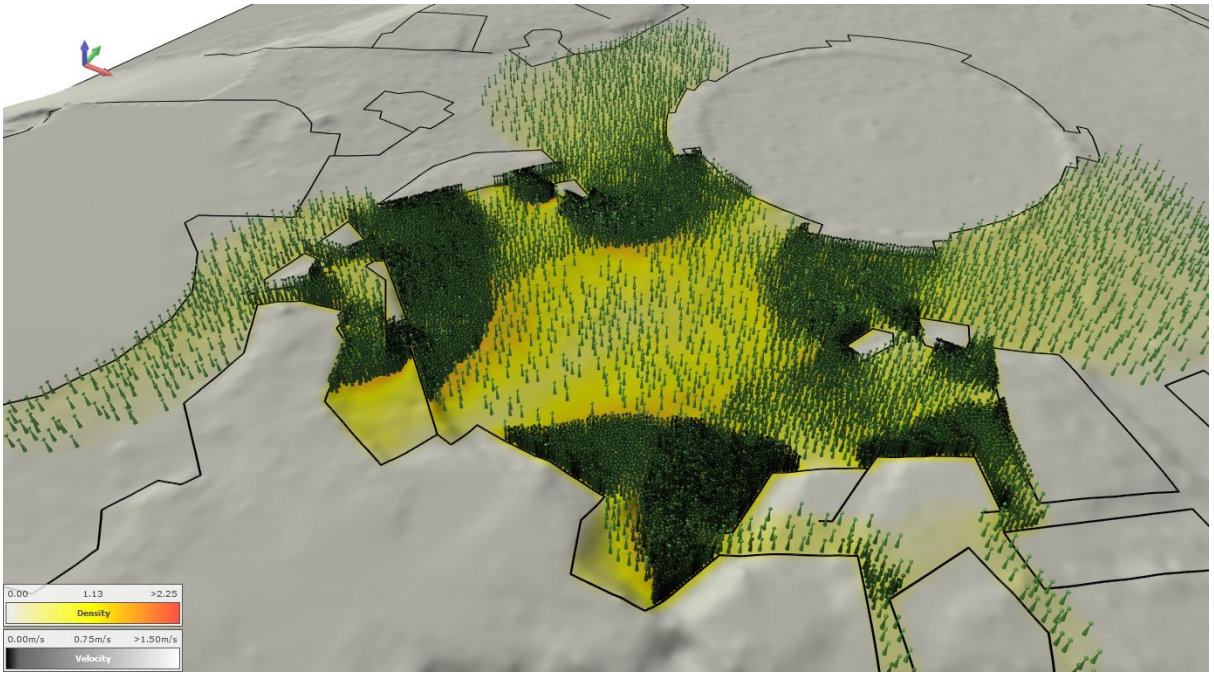
- Decals with fixed zoom behavior (zoom-independent size/thickness in screen space)
    - GPU-based polygon shifting required
- Glyphs on top of decal lines
- Railways (procedural texturing of stencil areas)



Topic: Crowd visualization for evacuation scenarios

Rendering of crowds is challenging because it requires to animate and display thousands of individuals (agents) with different characteristics and movements in a life-like manner. However, geometry, textures, and animations can often be reused and shared among multiple agents for efficient rendering. Additional data given by a simulation have to be visualized on the agents, such as their stress level and emotional state. In Visdom, we want to visualize the results of our existing simulation including

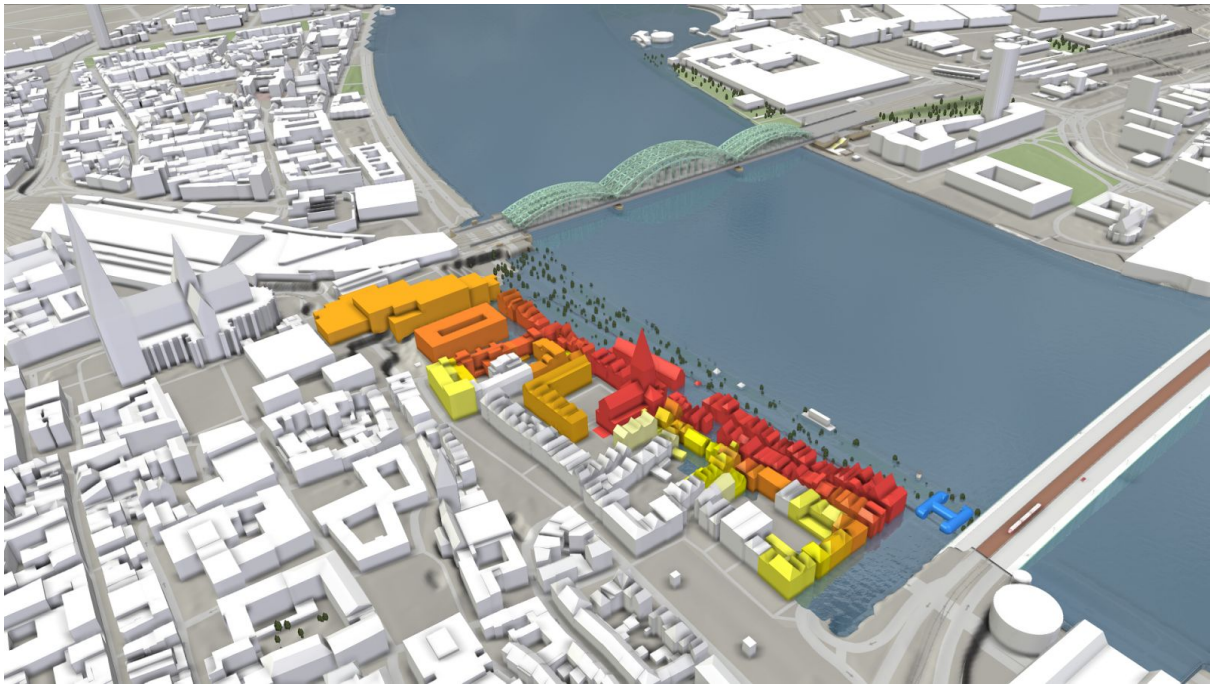
- Rendering of walking people, running people, standing people, desperate people, panic; derived from time step, position, velocity, density field
- People as glyphs
- Thought bubbles for people



Topic: WebGL Client for public communication

Interactive visualization of simulation data is very effective to communicate the risks of scenarios to the general public. Making the visualization browser-based removes the need for a dedicated viewer application and lowers the hurdle to interact with it. However, this means moving the rendering to the client side with possibly weak graphics card, so efficient rendering of only the most important information is crucial. As all data required for rendering have to be transferred to the client over the internet, an important task is to identify and extract the required subset of data from all data available to the server and stream it efficiently. The primary goals are:

- Exploit client-server architecture of Visdom to render scenarios on the client side using WebGL
- Basic rendering of terrain heightfields, meshes (buildings, barriers), water

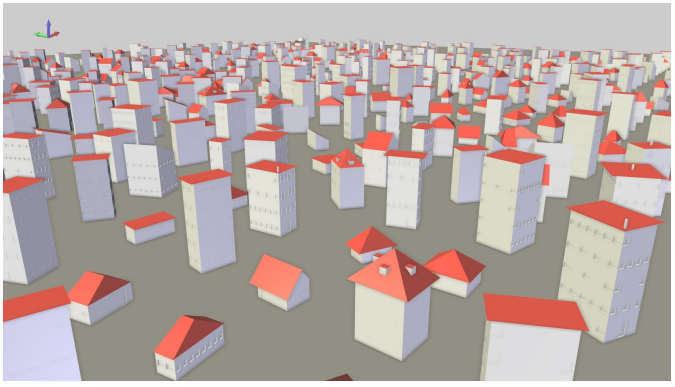


Topic: Integrating new simulations

- NVidia Flex
  - https://developer.nvidia.com/flex
  - Convert shape or CityGML data into particle-based flex representation and simulate earthquakes
  - Simulate hydraulic structures such as turbines using particles and couple with a large-scale flood simulation
  - Decouple cars from other, large-scale simulation and simulate collisions, e.g, in tunnels
- Traffic simulation
  - Based on Sumo http://www.dlr.de/ts/en/desktopdefault.aspx/tabid-9883/16931_read-41000/
  - Based on MatSim http://www.matsim.org/
- Wildfire
- Pedestrian simulation for evacuation scenarios
- Groundwater simulation

Topic: OpenData enhancements

- Buildings generation from shape files and satellite images
  - Generate roofs
  - Generate floors



Topic: CUDA implementations for expensive GIS algorithms

- Lines and polygons (unions, intersections, ...)

Topic: Visual analysis of user interaction in large data flows

- Identifiy user work flows
- Identify user errors and request chain causing the issues
- Use multiple linked info vis views to analyze requests