Server-Based Rendering of Large 3D Scenes for Mobile Devices Using G-Buffer Cube Maps

Juergen Doellner, Benjamin Hagedorn, Jan Klimke

Computer Graphics Systems Department
Hasso-Plattner-Institut
University of Potsdam, Germany

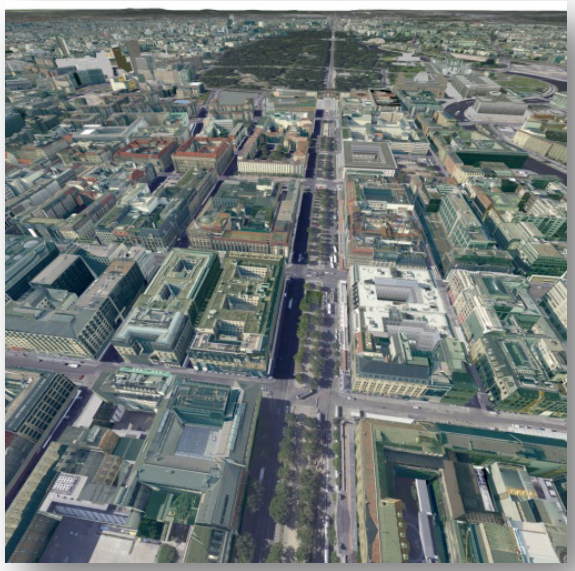
Web3D 2012, August 4th 2012

Motivation and Goals

Delivery of **high-quality 3D visualizations** of complex, massive 3D content in a way that allows for **interactive exploration** of a 3D geovirtual environment with only **moderate hardware requirements**.

3D visualization and massive **3D models on mobile devices**

Lower entry-barriers for users through **intuitive**, **touch-based interfaces** for complex 3D content (especially on mobile devices)

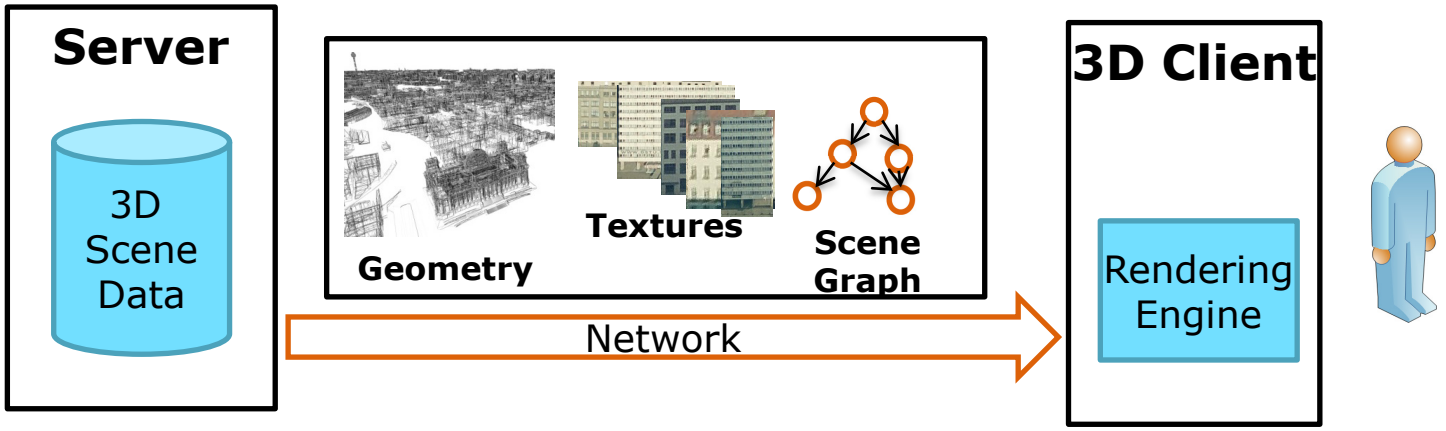


Complex 3D scene

Interactive, lightweight client application

Streaming Approaches for 3D Models

Common distributed visualization systems **transfer geometry, textures and scene graphs** to client applications



Large 3D scenes are *difficult to implement and operate* on mobile devices due to

- Limited network bandwidth / reliability
- Limited main memory
- Limited processing power
- Limited 3D rendering hardware and software capabilities

Image-Based Approach: G-Buffer Cube Maps

Server-side generation of **G-buffer cube maps** and transfer to 3D clients

- **3D scene approximation**
  - Discretized
  - Omni-directional
  - Viewpoint-dependent
  - Multi-layered (color, object id, depth, normals, …)

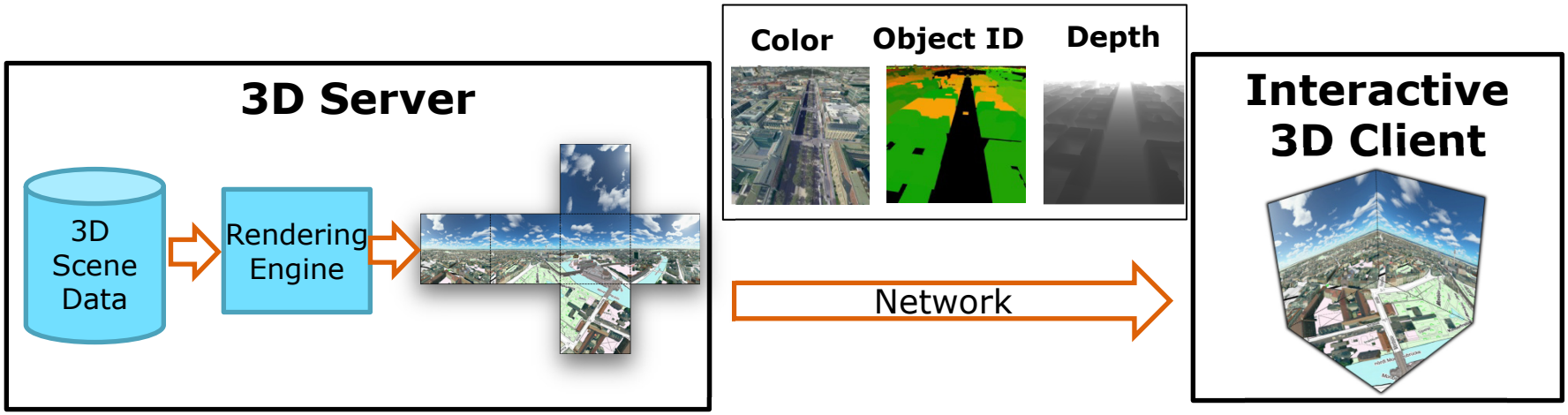
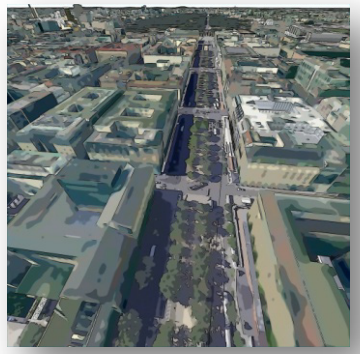
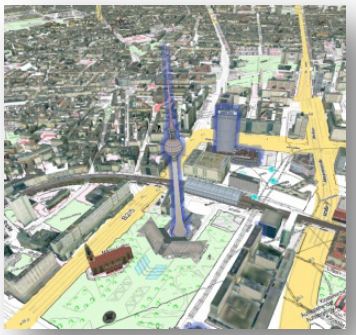


Image-Based Approach: Characteristics

- The complexity of G-buffer cube maps is independent from the 3D scene's complexity.
- 3D rendering within a **controlled server environment**
  - Known and tested server-side 3D graphics hardware and software
  - Stable implementation of advanced 3D rendering techniques (e.g., multi-pass rendering, shader programming)
  - Independent from client capabilities
- 3D Server interface is based on the Web View Service (WVS)
  - Standard proposal for 3D portrayal of the Open Geospatial Consortium (OGC)
  - Reusable, interoperable 3D visualization
- Original **3D content is kept protected**: 3D scene data never leaves the server environment.



NPR Effect

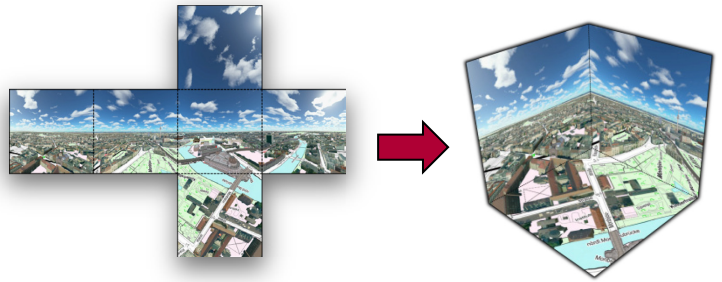


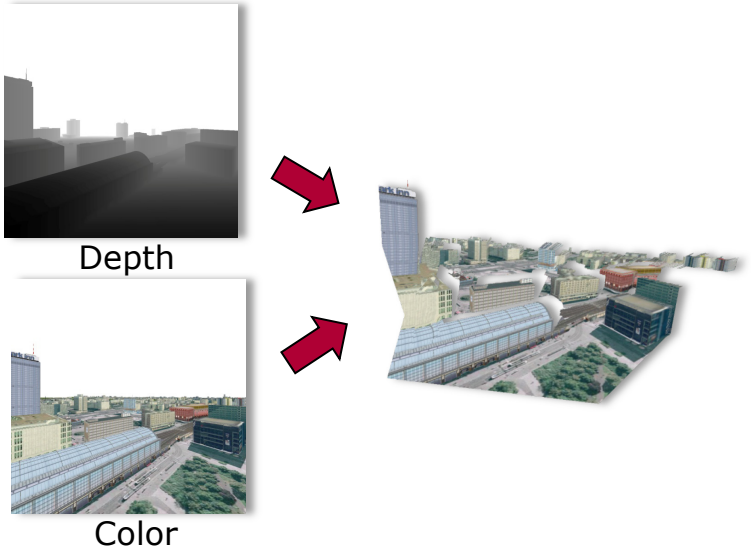
Highlighting effect

3D Client

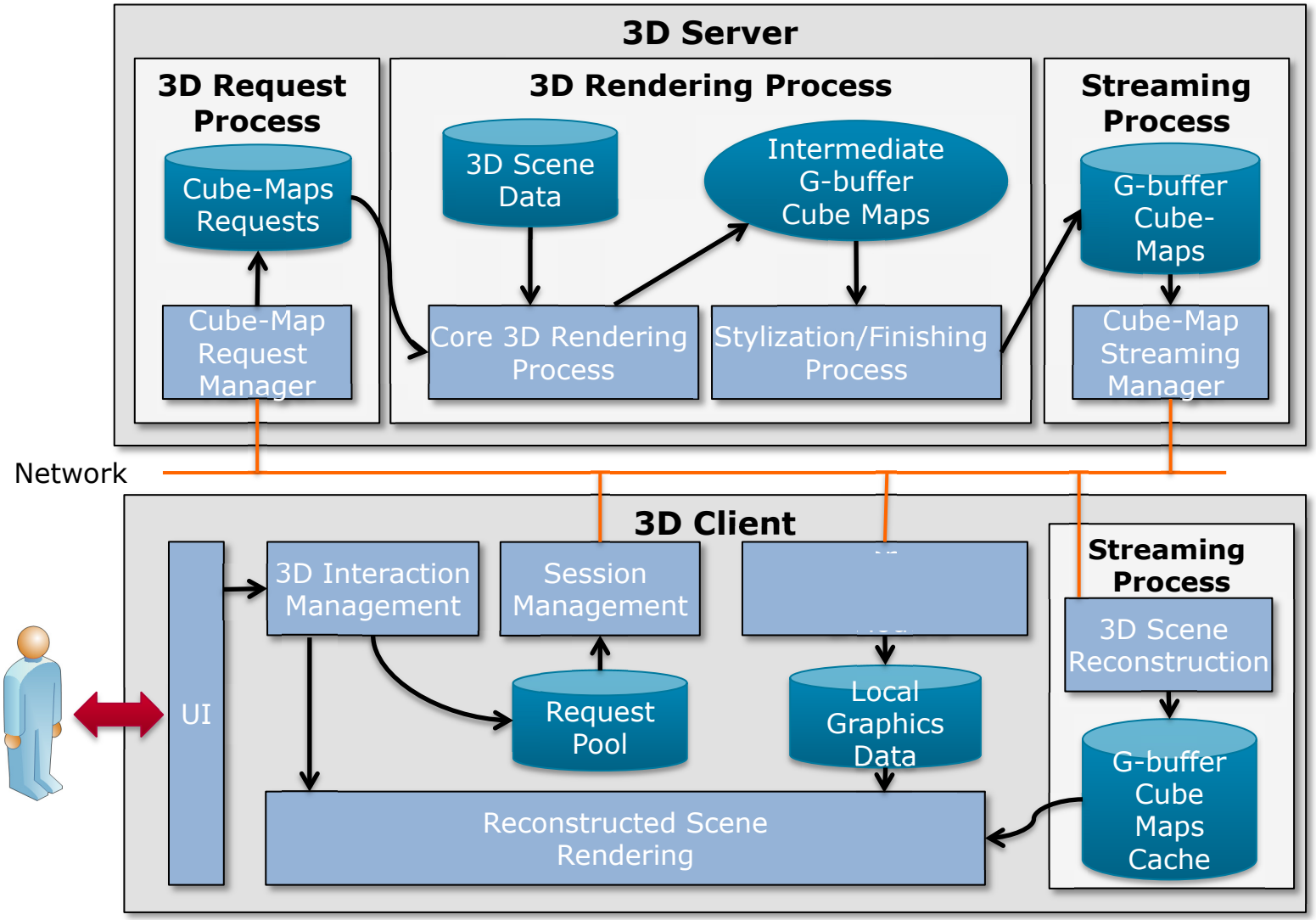
Interactive display of and user interaction with a reconstructed 3D scene from server generated G-buffer cube maps

1. **Cube map panorama** (12 textured triangles)
   - **High quality, omni-direcional** visual representation of large server-side 3D scene
   - Only valid for one **viewpoint**
2. **Geometry reconstruction** from depth information
   - Approximated representation during camera transitions
   - Allows a user to navigate uninterrupted through the 3D scene





System Overview



Case Study – Berlin 3D

- **Virtual 3D city model of Berlin**:
  - One of the worlds largest fully textured 3D city models
    - >550,000 Buildings,
    - > 4,000,000 facade textures,
    - > 350 high detail buildings
  - 3D city model as platform for data integration and presentation

http://www.businesslocationcenter.de

- **Use case: City marketing**
  - Expensive transport of rendering and storage systems for city model presentation on real estate fairs
  - 3D City model as iPad App for mobile presentation
  - Significantly lower costs through portable solution

- **Result:** Users **realize** the application as **"true 3D application"**

Video


Server-Based Rendering of Large 3D Scenes for Mobile Devices Using G-Buffer Cube Maps
Jürgen Döllner
Hasso-Plattner-Institut
Potsdam
doellner@hpi.uni-potsdam.de
Benjamin Hagedorn
Hasso-Plattner-Institut
Potsdam
benjamin.hagedorn@hpi.uni-potsdam.de
Jan Klimke
Hasso-Plattner-Institut
Potsdam
jan.klimke@hpi.uni-potsdam.de
HPI Hasso Plattner Institut
IT Systems Engineering | Universität Potsdam
www.hpi3d.de

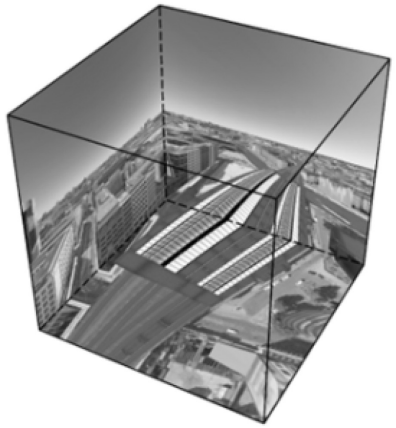
Conclusions and Future Work

- **Server-generated G-buffer cube maps**
  - Decouple server-side 3D scene complexity from transmitted data and significantly reduce complexity on client side
  - Facilitate interactive, high-quality visualization of large 3D scenes on mobile devices

- Server side rendering **facilitates development and distribution of high-end rendering techniques** since no end-user hardware/software need to be touched

- Users tend **to accept network delays** as well as small amounts of visual artifacts **as long as their interaction does not "get stuck"**

- **Simplicity and moderate hardware requirements** of client application makes plugin-free, browser-based **WebGL implementations** feasible



Jan Klimke

jan.klimke@hpi.uni-potsdam.de

Computer Graphics Systems Departement

Prof. Dr. Jürgen Döllner

www.hpi3d.de

Partners: