LGM
Laboratory for Computer Graphics and Multimedia

Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

CERN

Hybrid Rendering System for Particle Collision Experimental Data Visualization

Ciril Bohak

ciril.bohak@fri.uni-lj.si
lgm.fri.uni-lj.si

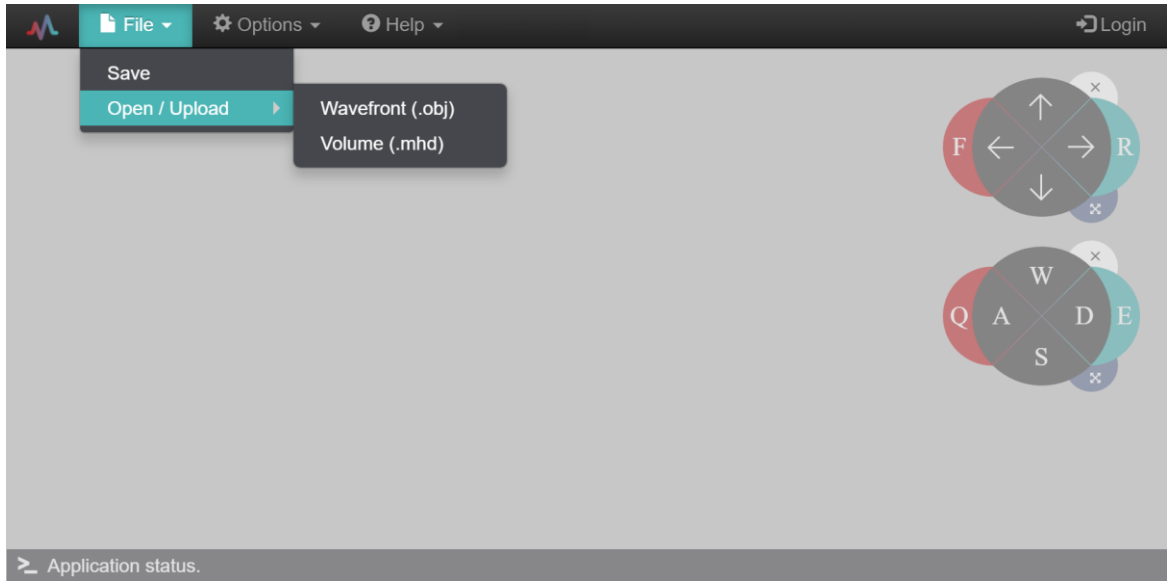
Naples – March 2018

Content

- Background;
- Problem Domains;
- Possible solutions;
- Hybrid Rendering System;
- Collaboration Options;
- Future Work;
- Conclusions;

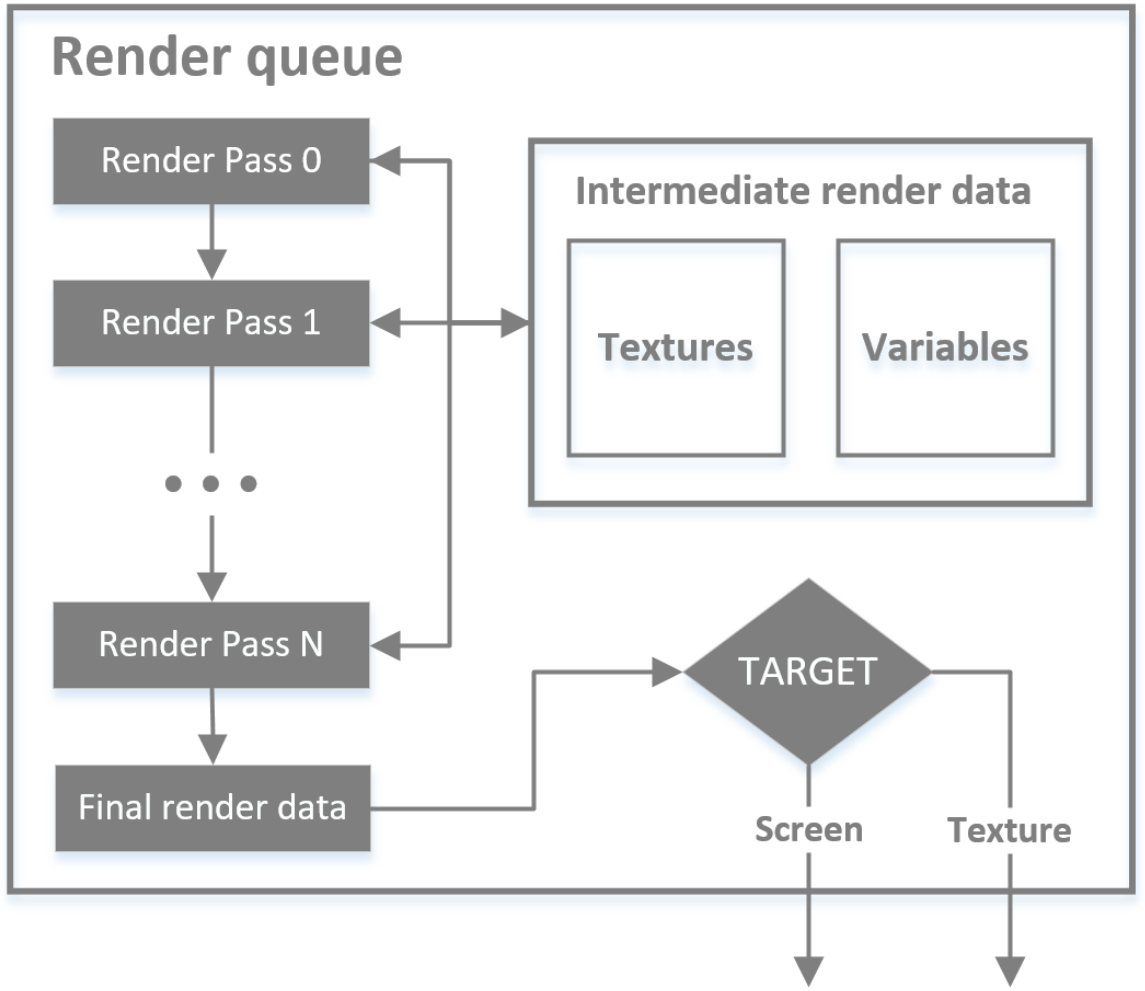
Background – Med3D

- Custom WebGL 2.0 based rendering pipeline;
- client-server oriented design (NodeJS);
- optional server-side data transformation;
- user annotations;
- user collaboration.



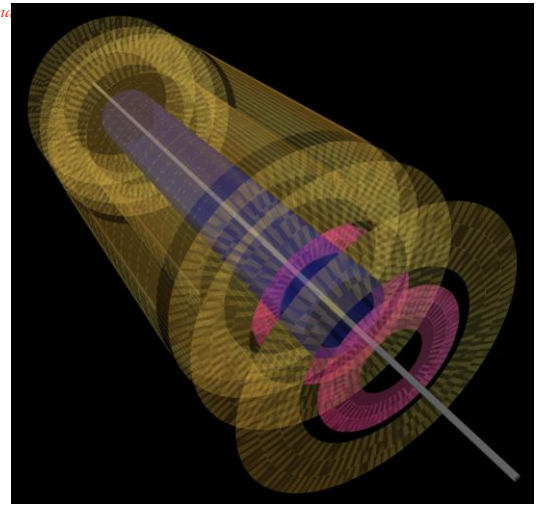
Background – Med3D - Custom pipeline

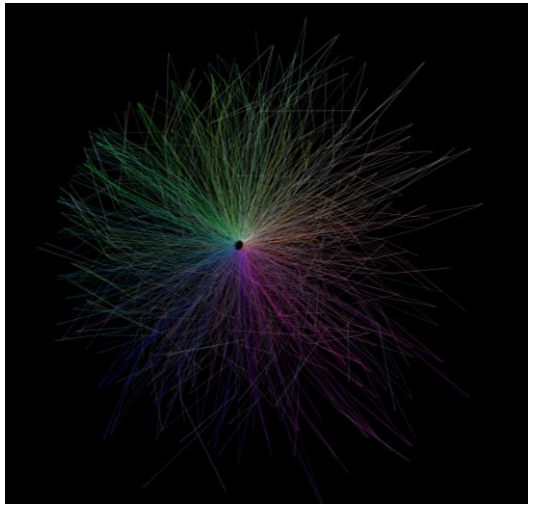
- Rendering pipeline designed based on deferred rendering pipeline;
- multi pass rendering support for option of combining multiple types of visualizations such as:
  - data rendering + view aligned annotations;
  - mesh rendering + direct volumetric rendering;
  - data rendering + flow visualization;
- option of customizing of scene rendering for individual user during sharing session;
- option for storing output images to single or multiple images.

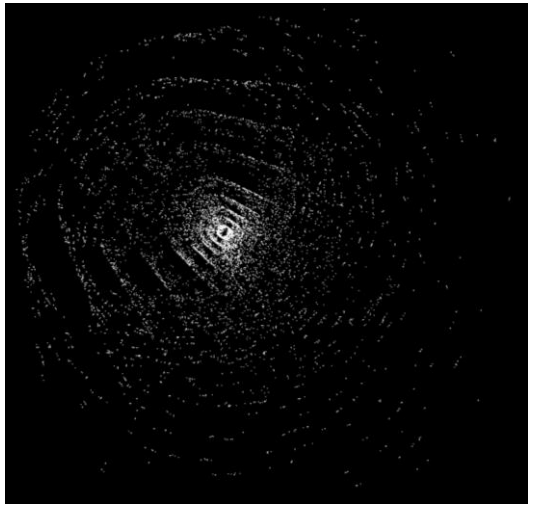


Problem domain

- Rendering geometry:
  - rendering experiment geometry;
  - not such a big problem;
- Rendering lines:
  - current experiments: ~ 10k lines;
  - future experiments: ~ 200k lines;
- Rendering points:
  - current experiments: ~ 100k points;
  - future experiments: ~ 5M points.







Possible solutions

- Client-side rendering system:
  - Pros:
    - high interactivity;
    - highly responsive;
    - possible off-line work.
  - Cons:
    - high-end hardware for complex visualizations;
    - a lot of data transfer to user;
    - browser limitations.

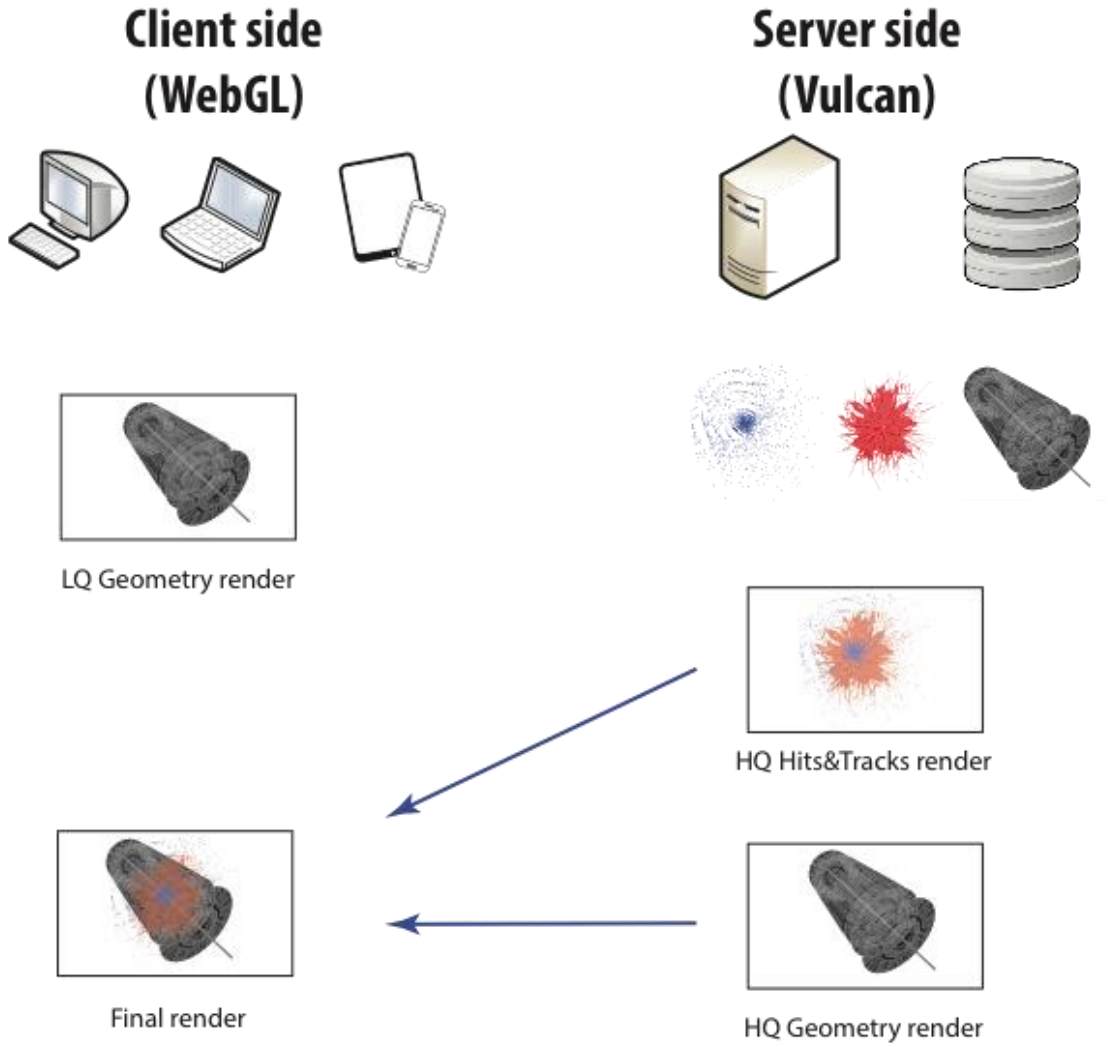
- Server-side rendering system (remote rendering):
  - Pros:
    - low-end hardware on client side;
    - less data transfer;
    - possibility of data protection.
  - Cons:
    - low interactivity;
    - response depends from network latency;
    - requires server-side system resources per user;
    - no off-line work.

Hybrid Rendering System

- Combining client-side and server-side rendering capabilities;
- client-side rendering of basic geometry;
- server-side rendering of complex geometry and event data (hits, trajectories, etc.);
- high interactivity with basic geometry;
- incremental rendering of more complex geometry and event data;
- optional selection of where to render what data.

- Pros:
  - low-end hardware on client side;
  - high interactivity;
  - less data transfer;
  - possibility of data protection.
- Cons:
  - slower high-quality rendering;
  - requires server-side system resources per user;
  - browser limitations;
  - no off-line work.

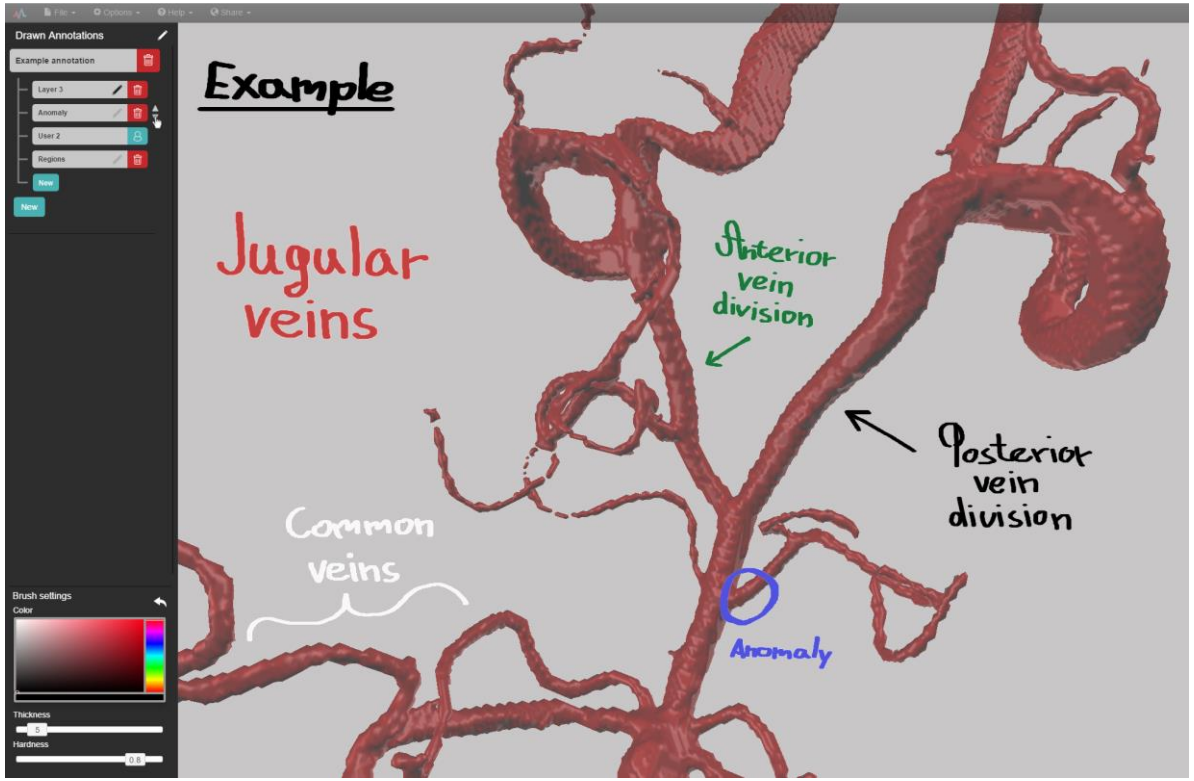
Hybrid Rendering System – Architecture



Collaboration Options – User annotations

two types of annotations:

- 3D location based annotations:
  - anchored to a specific point of interest on the model;
  - draggable in the view;
  - can be hidden or displayed;
- view aligned sketch based annotations:
  - aligned to a specific camera view;
  - multi-layered annotations;
  - user defined thickness, color and brush hardness.



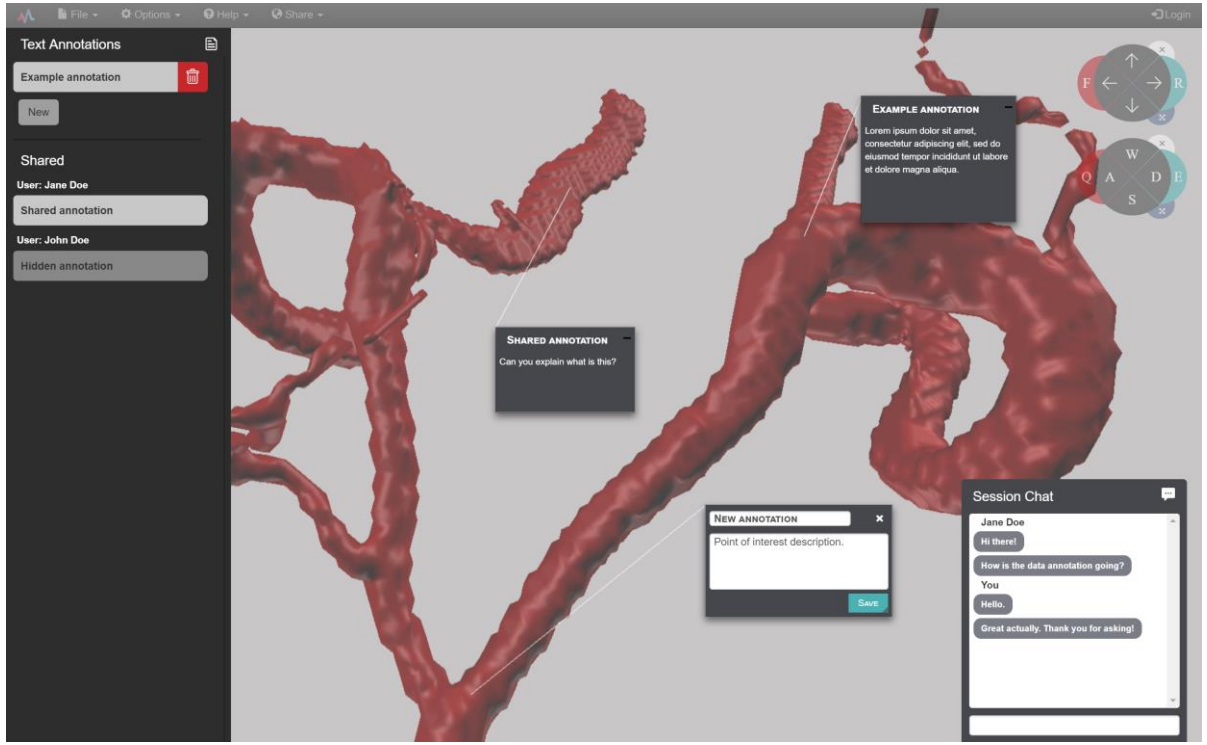
Collaboration Options – User collaboration

multiple modalities of user collaboration:

- text chat;
- data sharing;
- view sharing;
- annotation sharing.

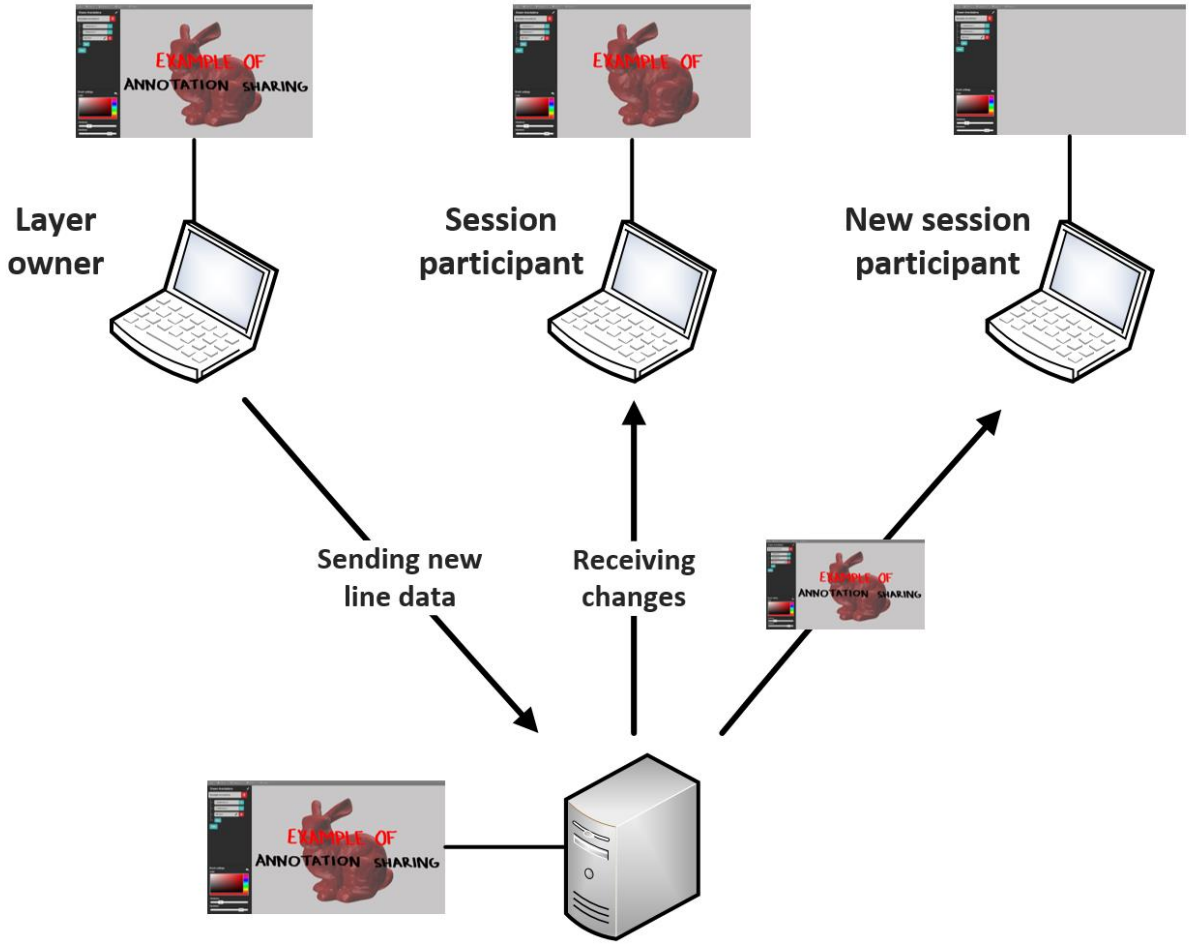
Med3D future plans:

- voice and video chat sharing;
- user management.



Collaboration Options – Sharing

- data can be stored at user's computer or on the server;
- collaboration is done through server:
  - the host sends scene description to the server;
  - the host can decide what to share (data, view, annotations or everything);
  - all the scene updates are sent to the server;
  - other users may connect to server at any point and get the current scene description;
  - all the updates are continuously sent to all the users.



Future work

- Use of physically based rendering techniques for visualizing the experiment geometry;
- use of complex visualization techniques for event data;
- support for different kinds of data;
- visualization of user uploaded data;
- scene properties saving;
- image saving;
- server side rendering jobs.

Conclusions

- Possibility of running visualization system on low-end devices (including mobile);
- less data transfer;
- option of high-quality rendering;
- offers high scalability.

Laboratory for Computer Graphics and Multimedia

Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

CERN

???

Questions?

Research proposal document.