DynCam: A Reactive, Distributed Point-Cloud Pipeline

Christoph Schröder

University of Bremen, Germany

cgvr.informatik.uni-bremen.de

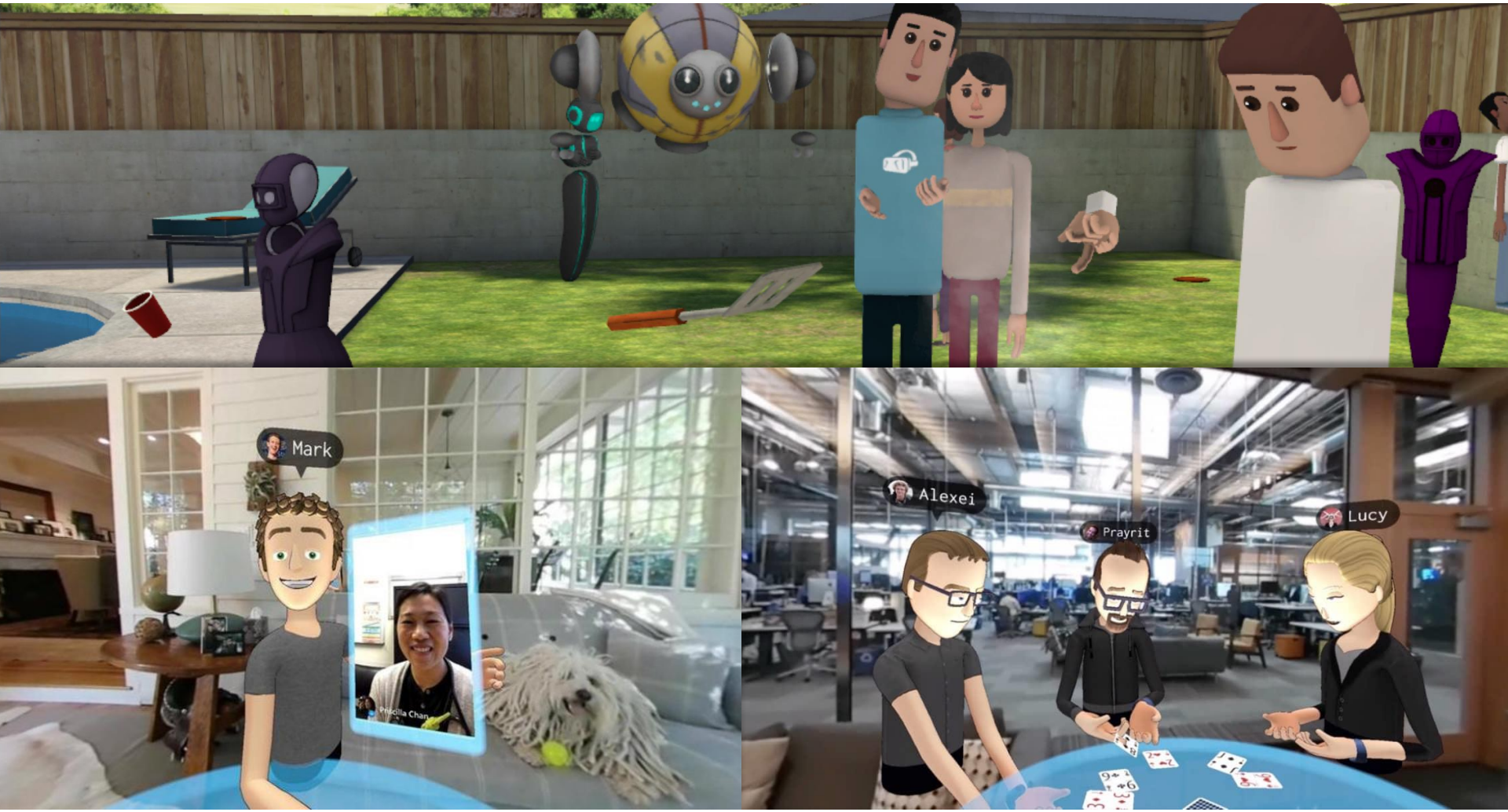
VRIC, April 2018

Telepresence

Telepresence

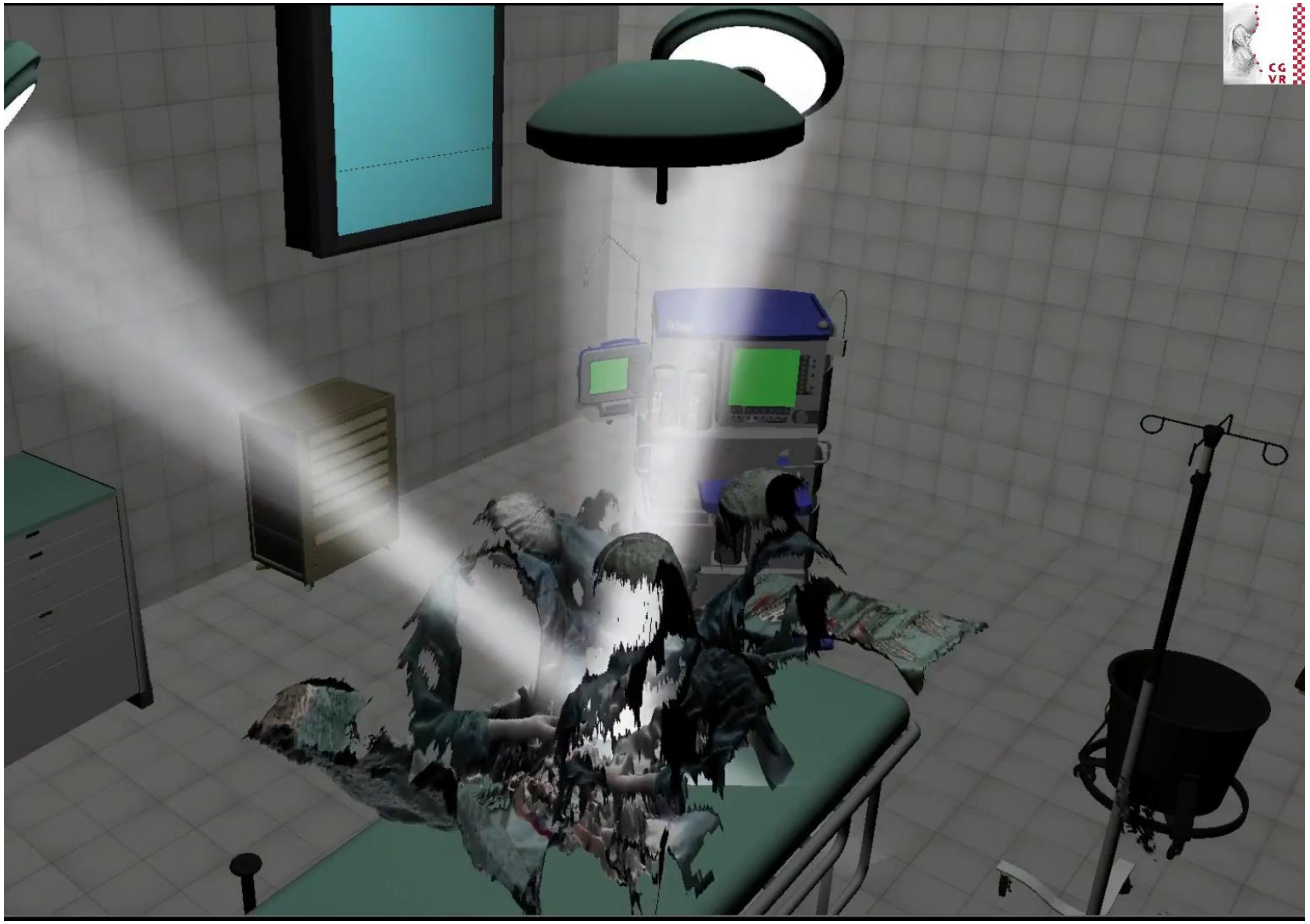
Altspace VR

Telepresence



Vision

Collaborative Virtual Environments



Our Longterm Goal

- Requirements
  - Realistic visualization
  - Shared virtual environment
  - Precise interaction
- Technologies
  - Multiple 3D cameras
  - Merge point clouds
  - Realistic rendering

Related Work

- Fusion Kit[1]
- Fusion4D[2]

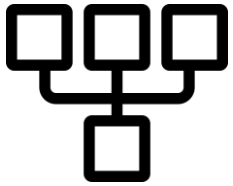
- ROS (Robot Operating System)[3]

[1] Michael Rietzler, et al. 2016. *FusionKit*. ACMSIGCHISymp. 8 - EICS ’16
[2] Mingsong Dou, et al. 2016. *Fusion4D*. ACMTrans. Graph. 35
[3] Morgan Quigley,et al. 2009. ROS: an open-source Robot Operating System. In ICRA Workshop on Open Source Software

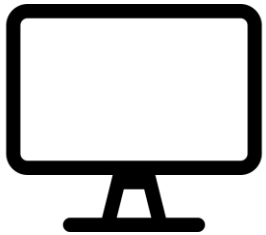
Our Contribution

- Framework for distributed point cloud processing
  - Easy-to-extend C++ pipeline
  - Implicit paralellisim
  - Different camera sources
  - Platform independent
- Basic visualization in Unreal Engine 4
- Improved end-to-end latency measurement method

Input



Pipeline

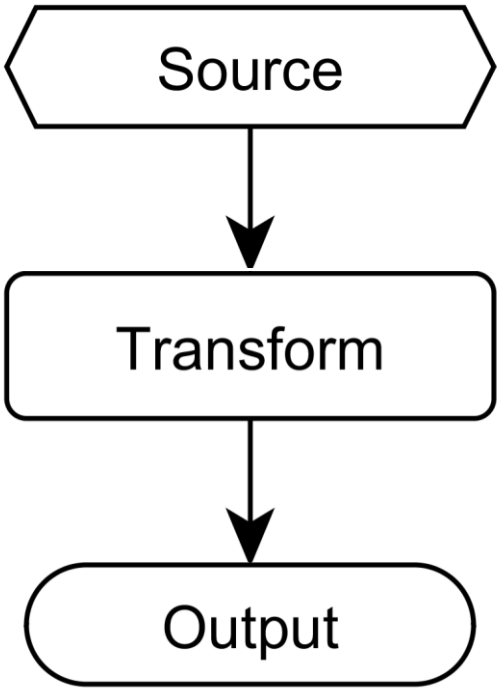


Visualization

Latency

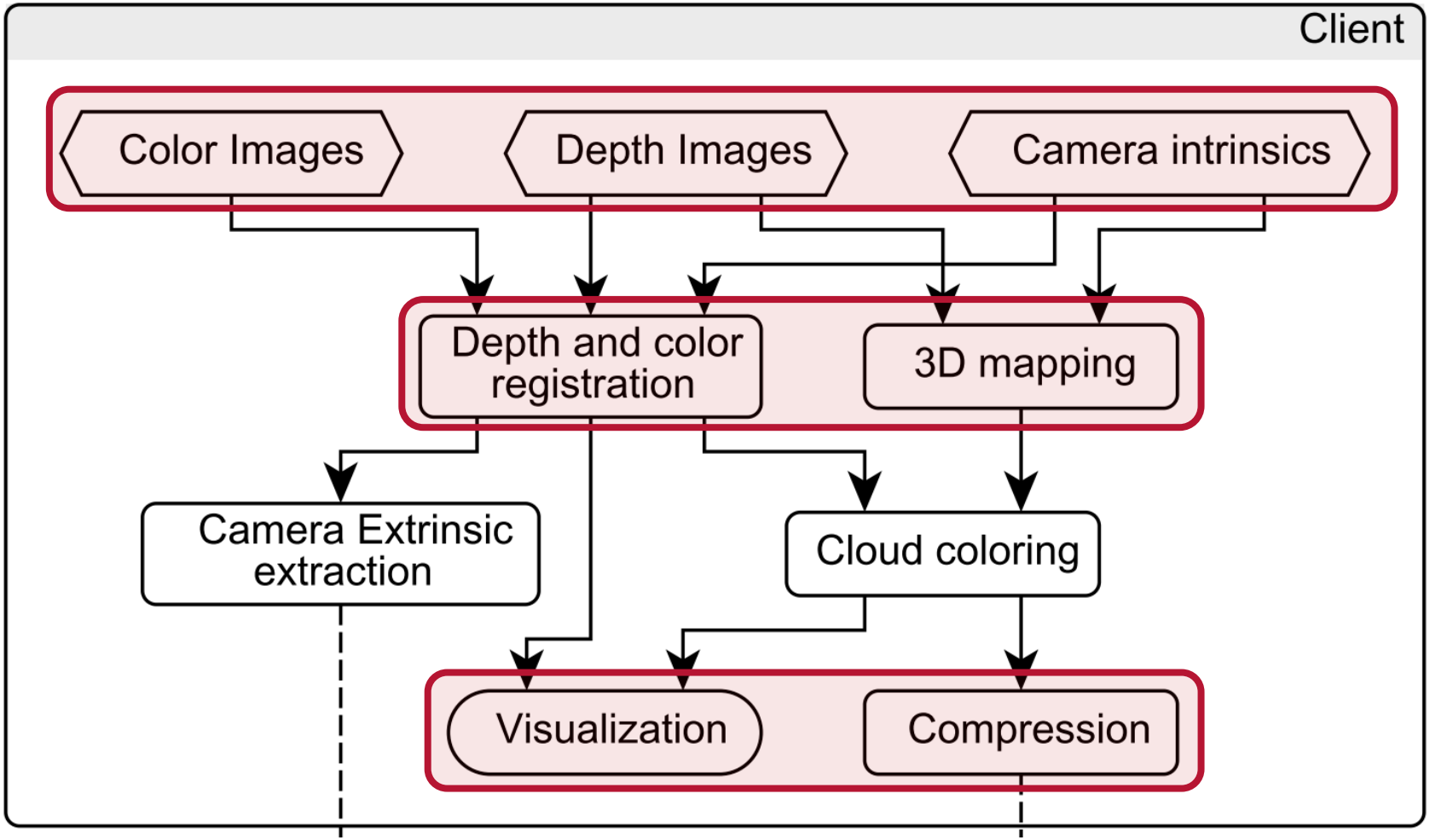
Reactive Architecture

- Updates
  - Automatic
  - Lazy
  - Implicitly parallel

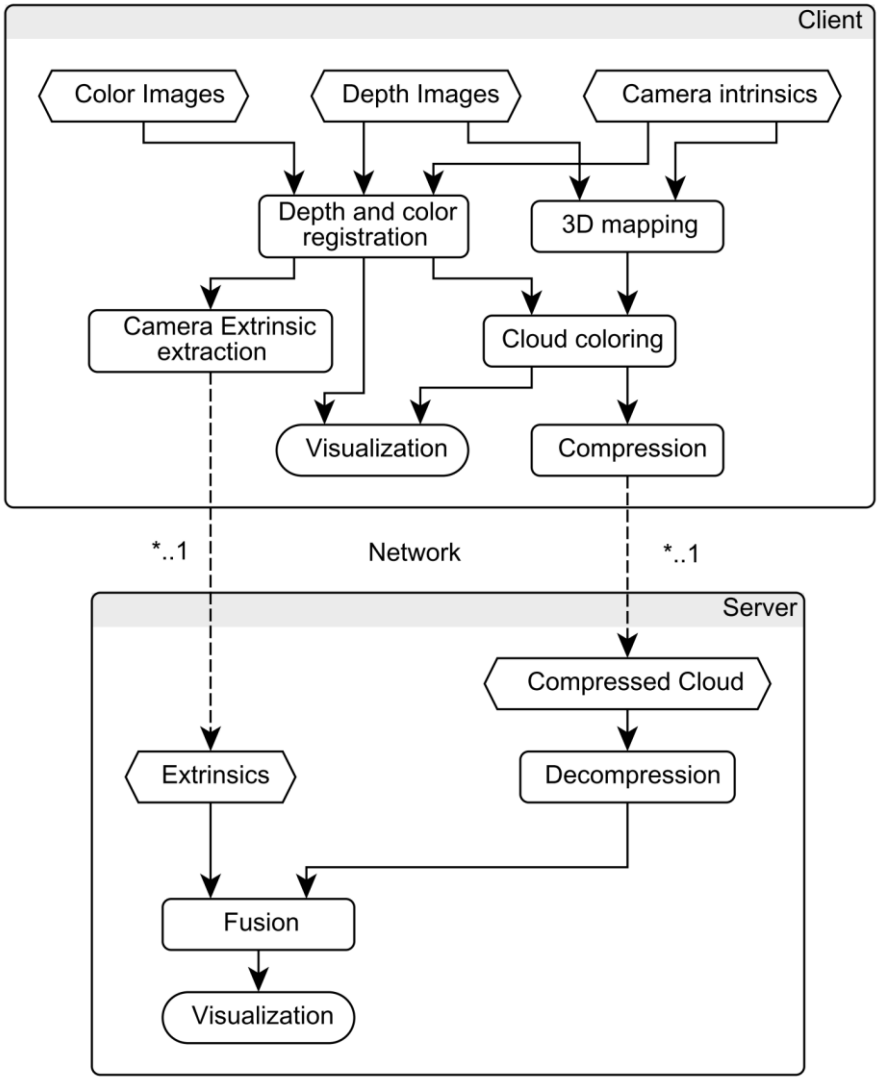


Conal Elliott & Paul Hudak. 1997. *Functional reactive animation*.
ACM SIGPLAN international conference on Functional programming - ICFP '97

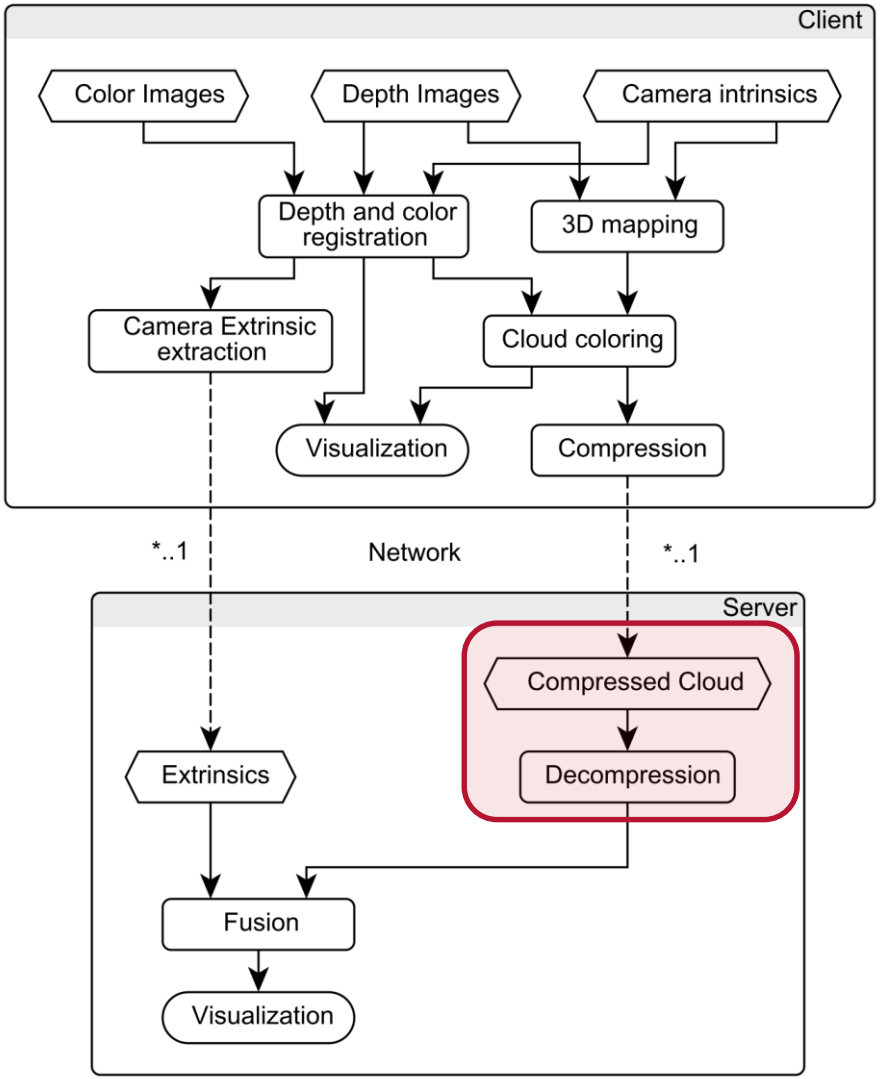
Reactive Architecture



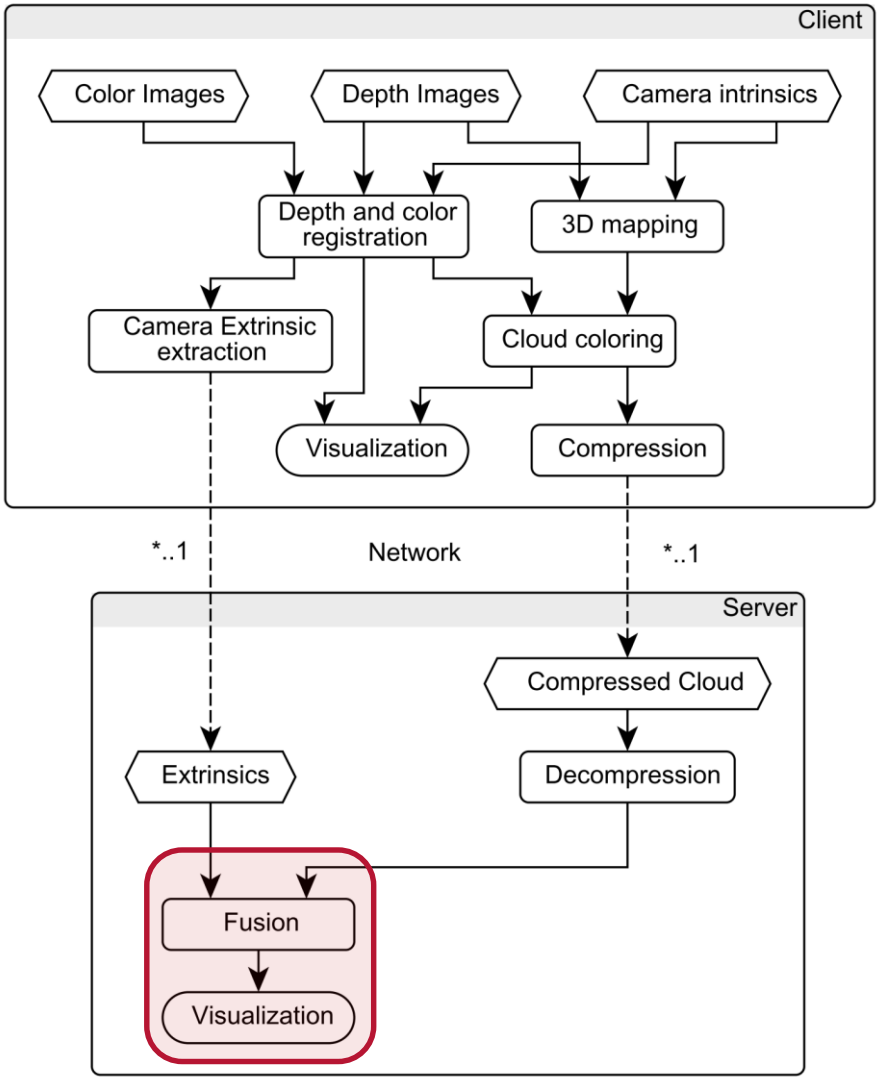
Reactive Architecture



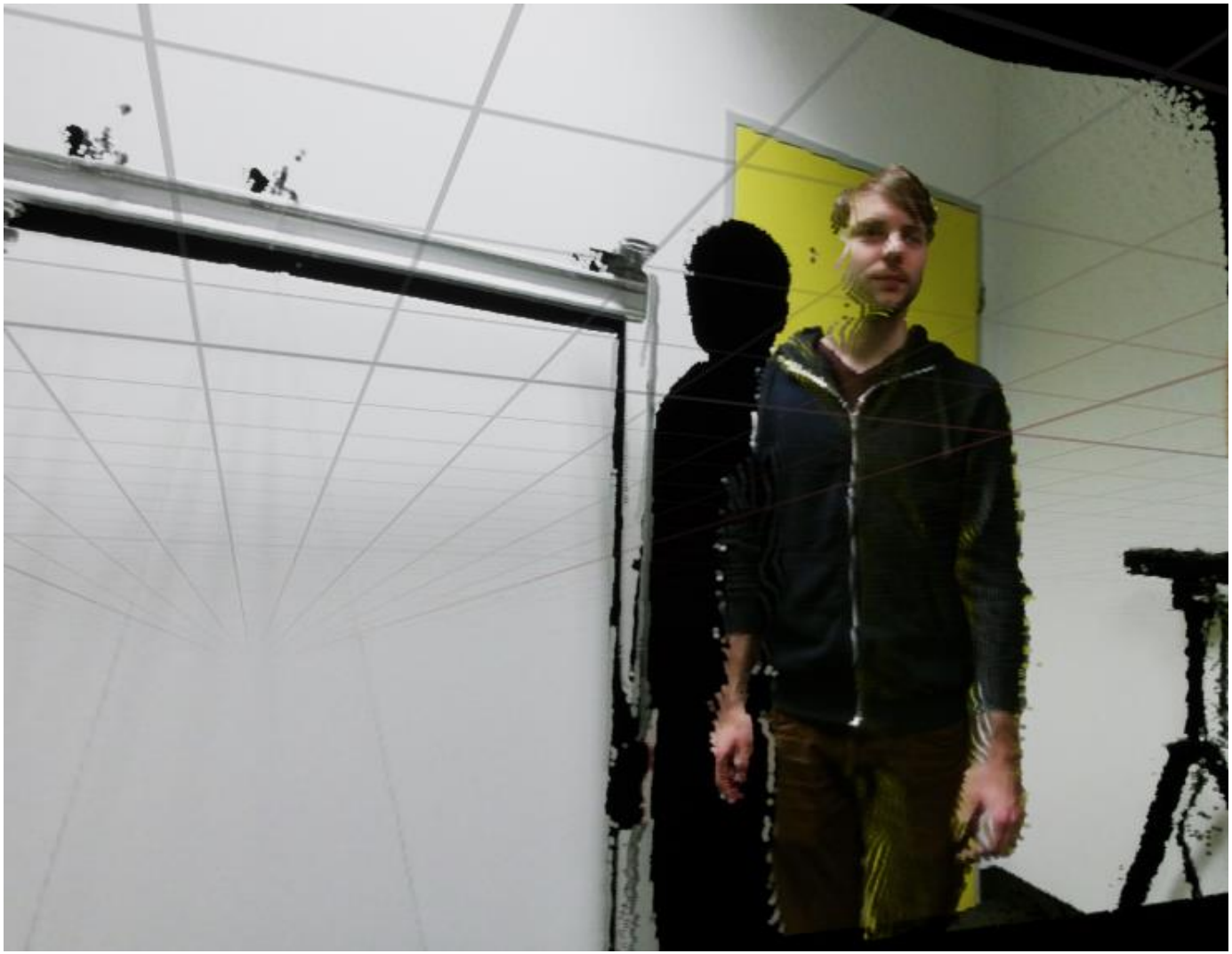
Reactive Architecture



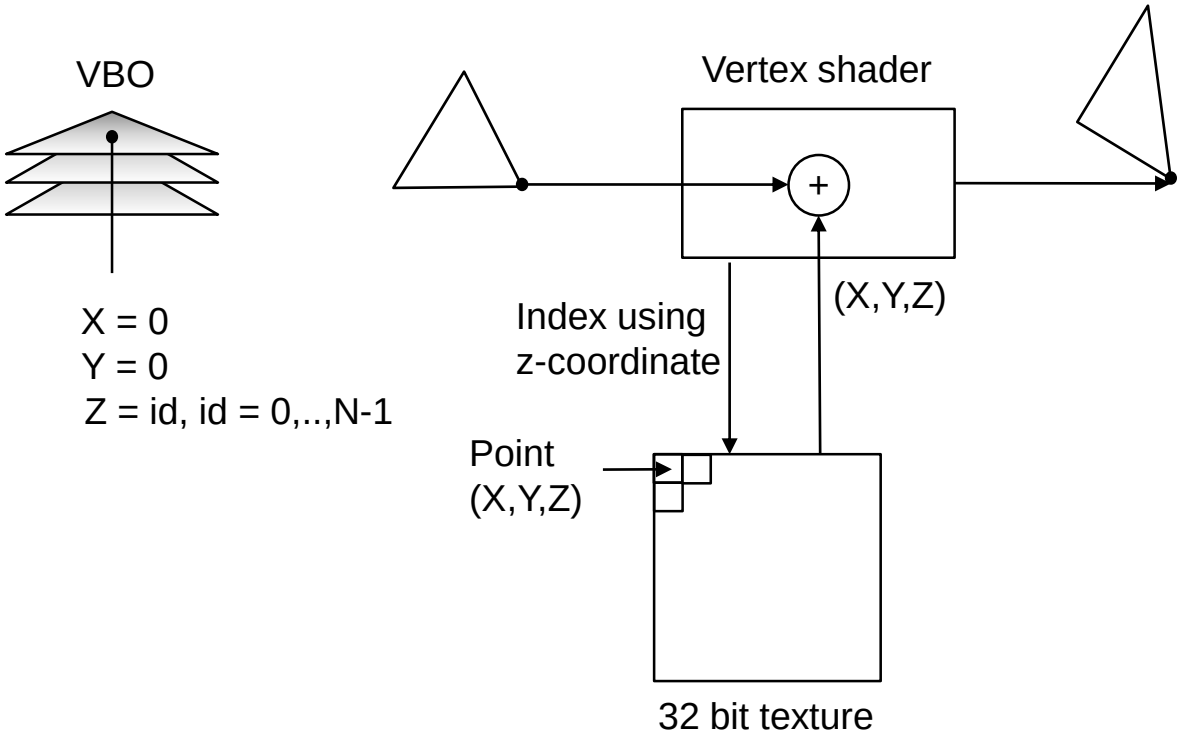
Reactive Architecture



Unreal Engine Integration



Unreal Engine Integration



Latency Measurment

- Based on method by Anthony Steed[4]

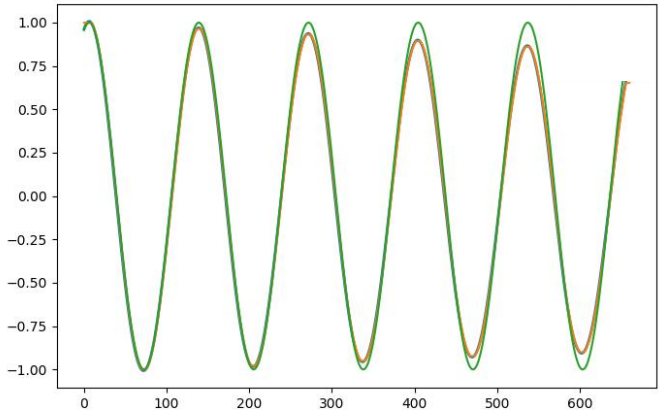
[4] Anthony Steed. 2008. A simple method for estimating the latency of interactive, real-time graphics simulations. In Proc. 2008 ACM Symp. Virtual Real. Softw. Technol. - VRST'08.

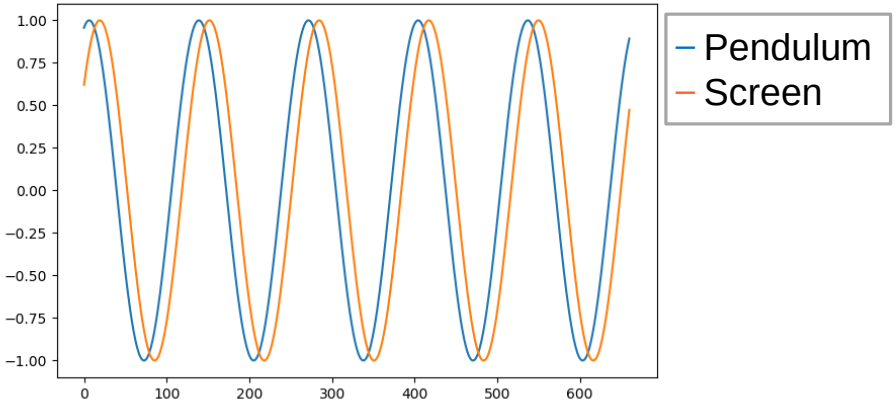
Latency Measurement

- Converted to Python
- Extract LEDs with OpenCV blob detection
- Outlier removal (moving average)
- Estimate frequency and phase angle with fft
- Optimize via curve fit

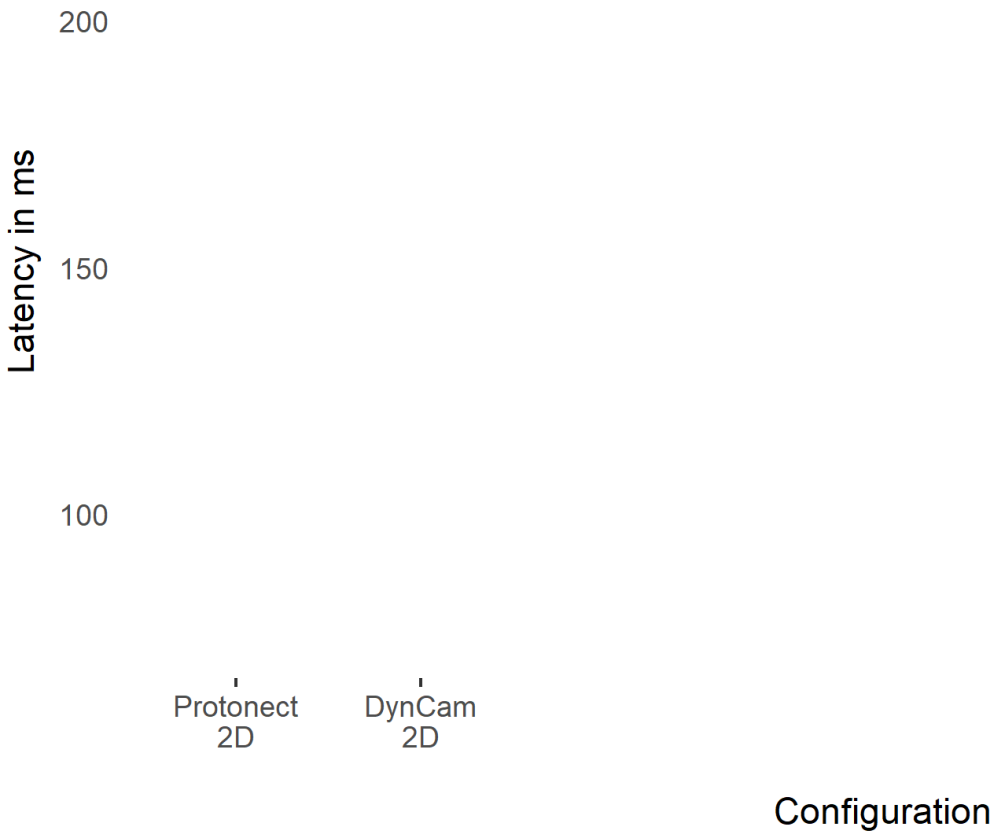
$$y(x) = A\sin(2\pi f x + \Phi)$$

- Latency $\Delta t = \frac{\Delta\Phi}{2\pi f r}$

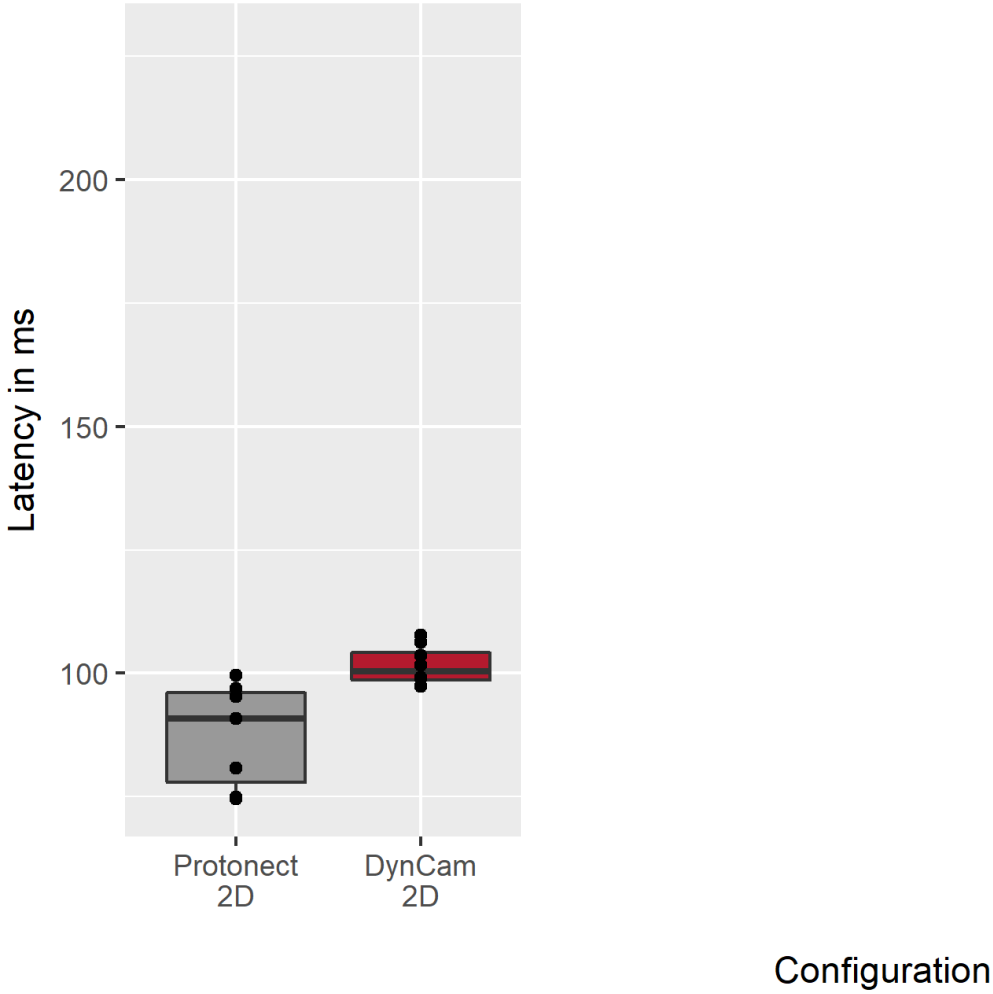




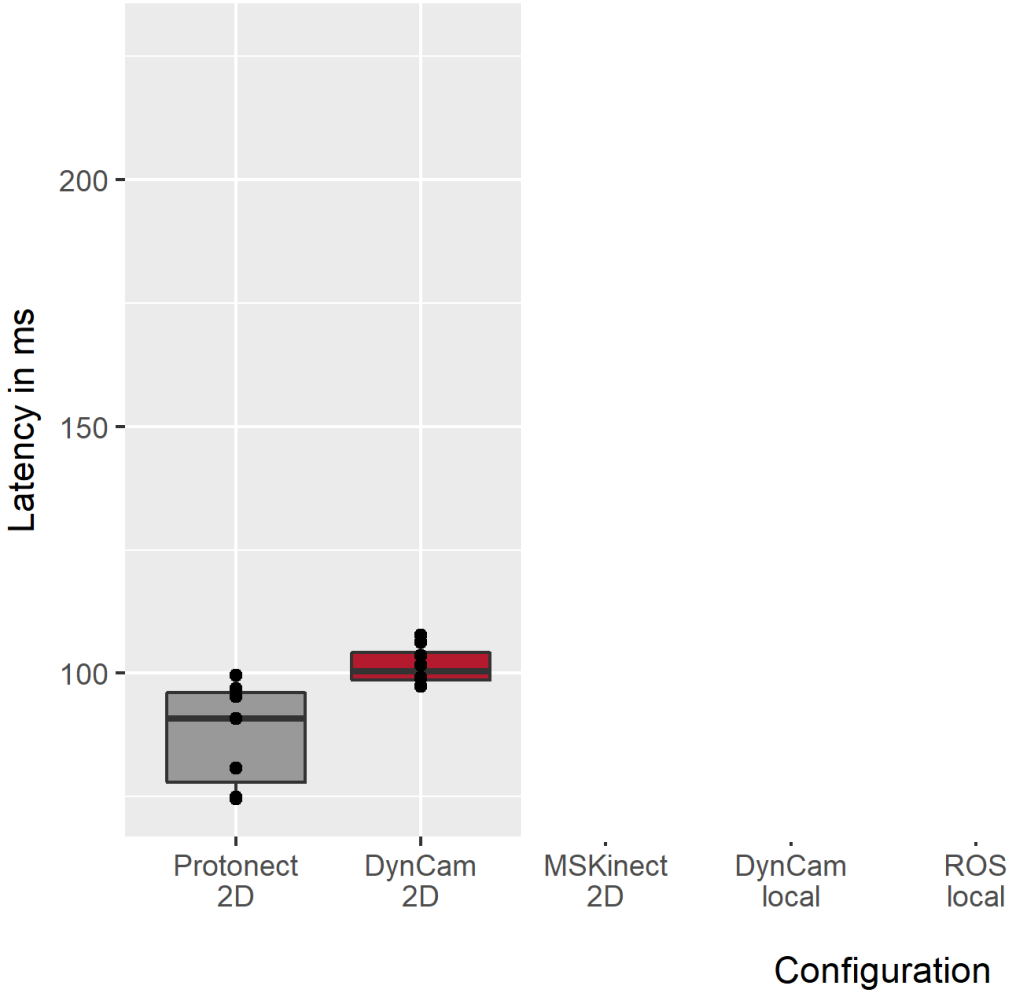
Latency


200
150
100
Latency in ms
Protonect 2D
DynCam 2D
Configuration

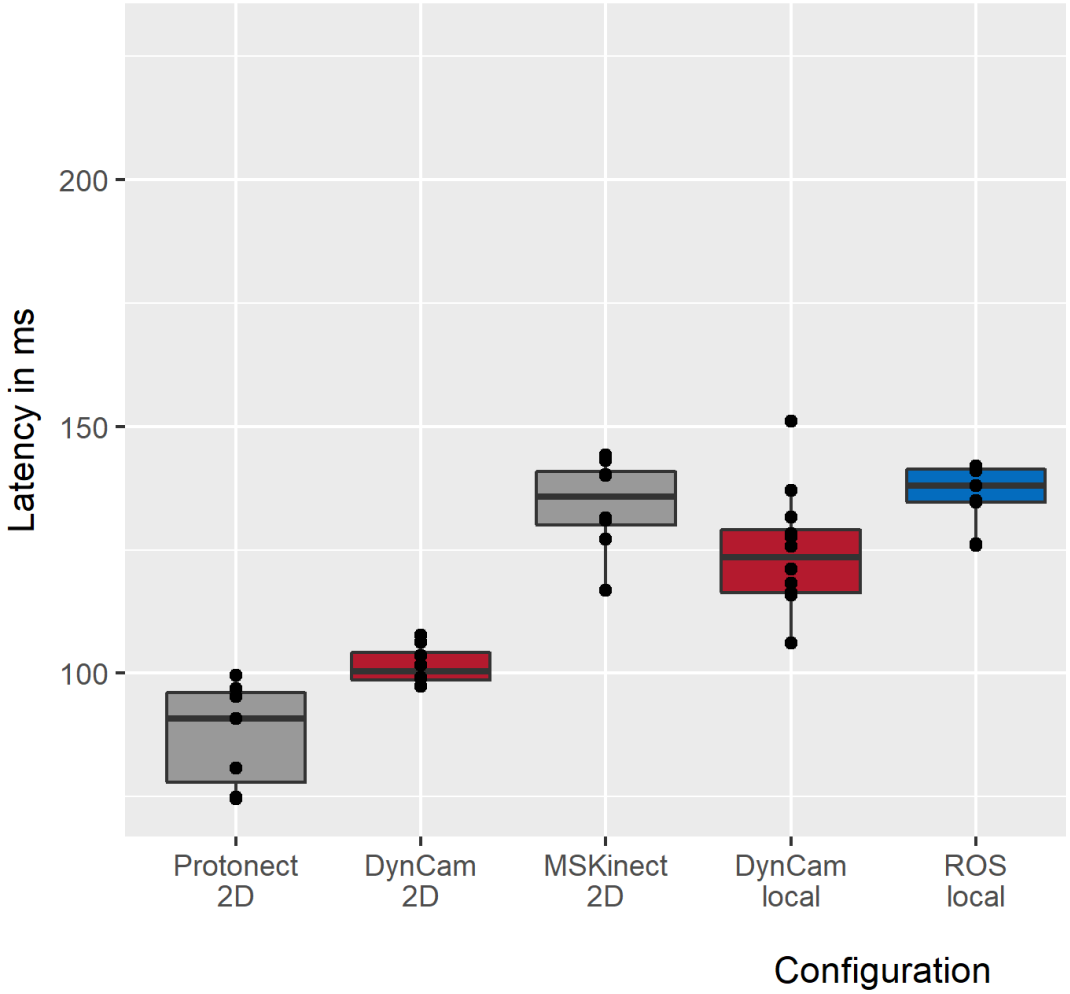
Latency


Latency in ms
200
150
100
Protonect 2D
DynCam 2D
Configuration

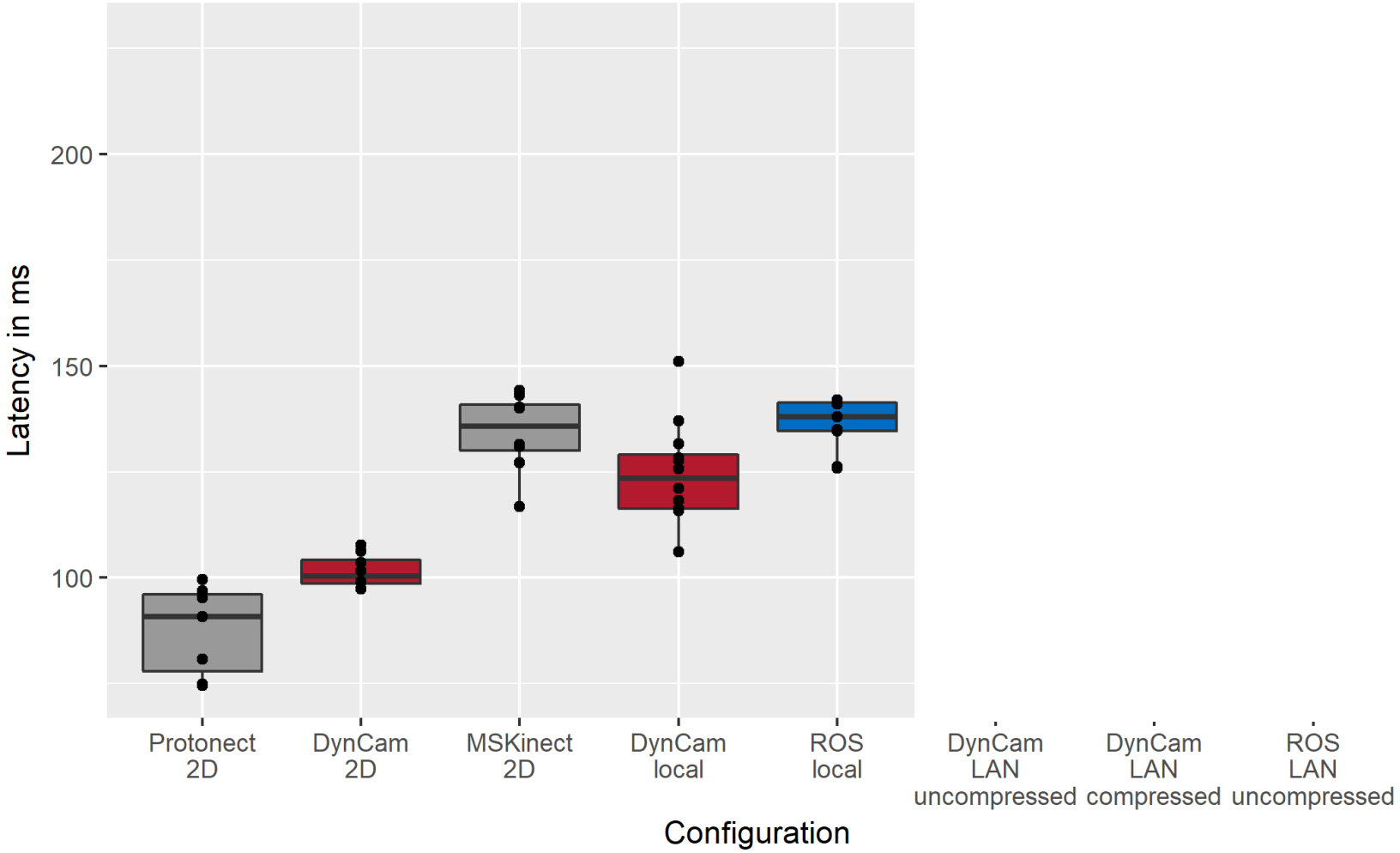
Latency


Latency in ms
200
150
100
Protonect 2D
DynCam 2D
MSKinect 2D
DynCam local
ROS local
Configuration

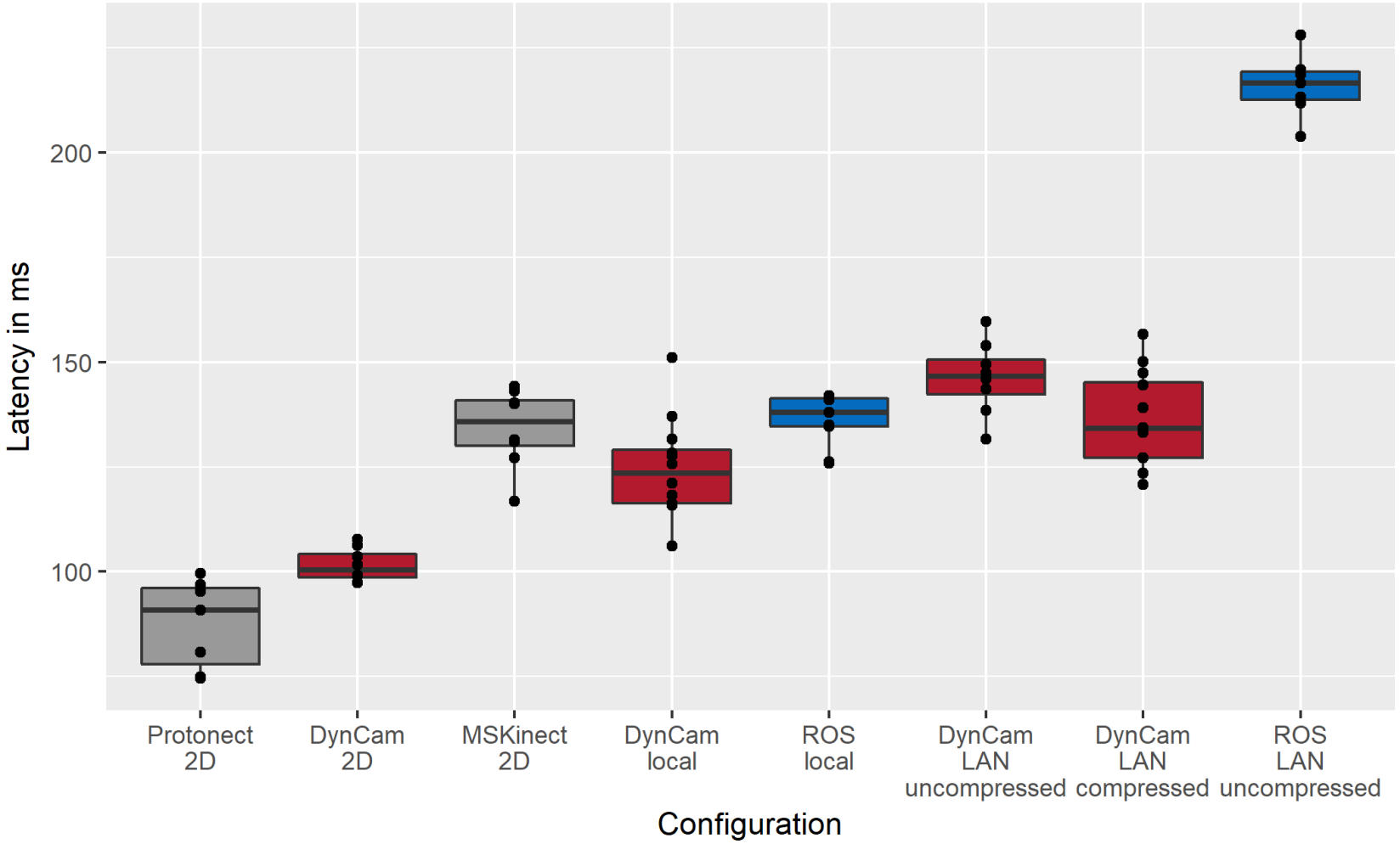
Latency


Latency in ms
200
150
100
Protonect 2D
DynCam 2D
MSKinect 2D
DynCam local
ROS local
Configuration

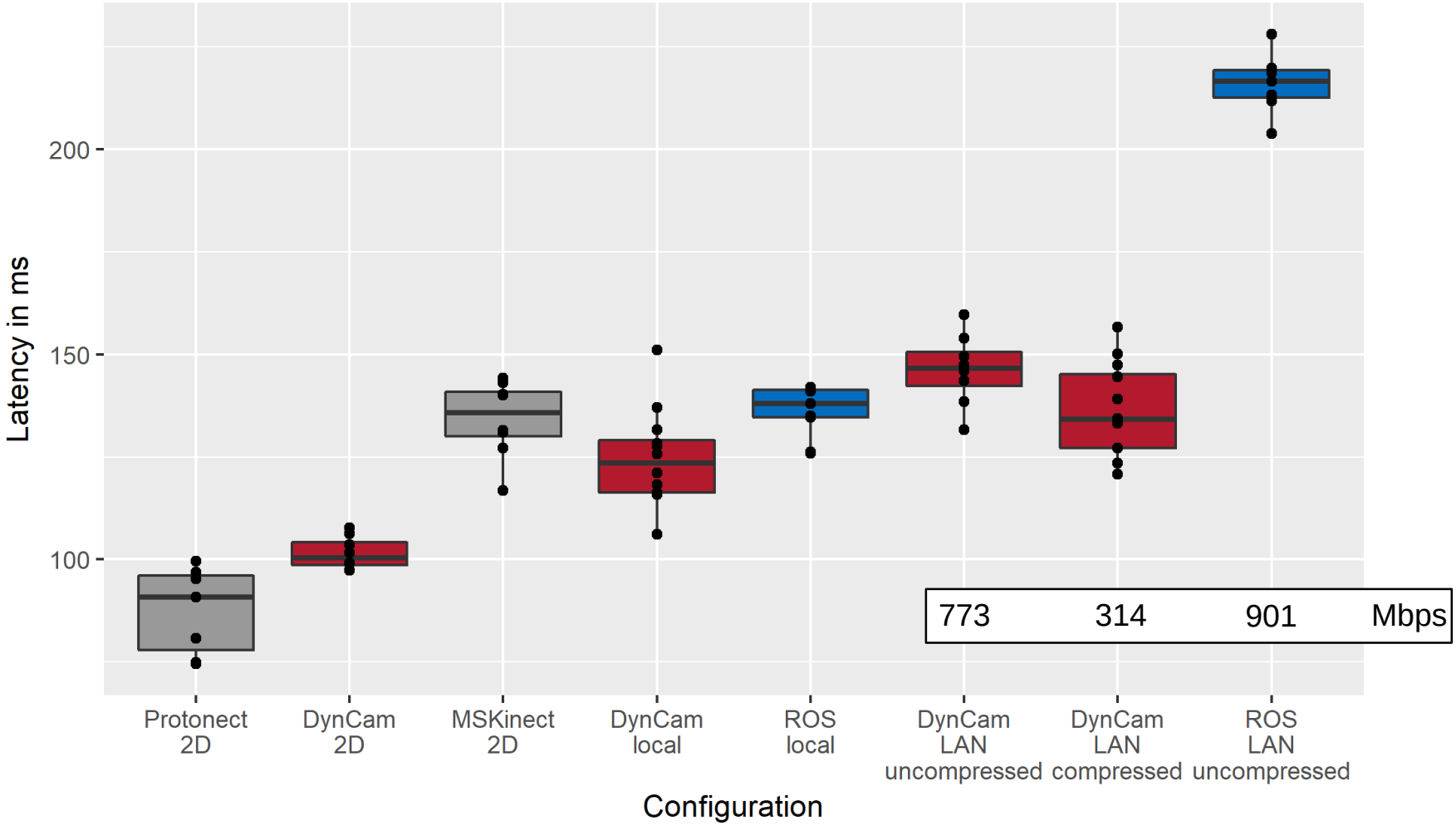
Latency


Latency in ms
200
150
100
Protonect 2D
DynCam 2D
MSKinect 2D
DynCam local
ROS local
DynCam LAN uncompressed
DynCam LAN compressed
ROS LAN uncompressed
Configuration

Latency

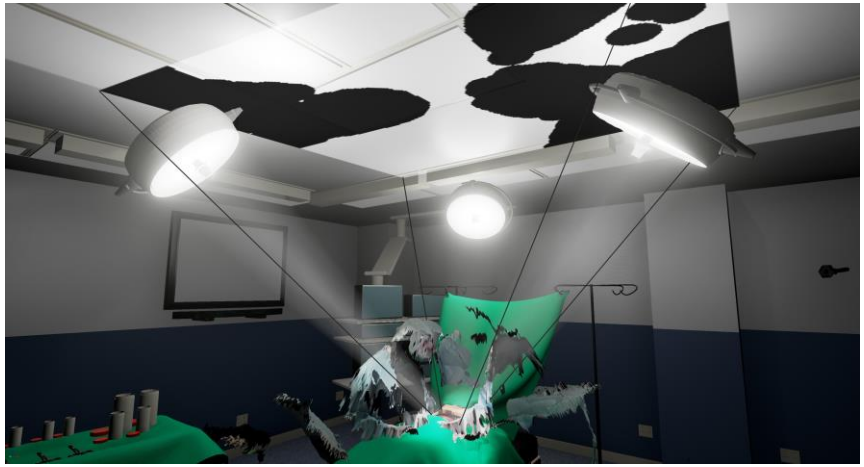

Latency in ms
200
150
100
Protonect 2D
DynCam 2D
MSKinect 2D
DynCam local
ROS local
DynCam LAN uncompressed
DynCam LAN compressed
ROS LAN uncompressed
Configuration

Latency


Latency in ms
200
150
100
773
314
901
Mbps
Protonect 2D
DynCam 2D
MSKinect 2D
DynCam local
ROS local
DynCam LAN uncompressed
DynCam LAN compressed
ROS LAN uncompressed
Configuration

Conclusion

- Distributed point cloud library
  - Platform independent
  - Implicit paralellisim
  - Easy to extend
  - 30% lower latency than ROS
- Unreal Engine integration
- Improved latency measurements
- Code
  - http://cgvr.cs.uni-bremen.de/papers/vric2018/



Conclusion and Future Work

- Enhanced rendering as splats or surface
- Improved compression
- Temporal resolution enhancement
- Automatic VR/AR alignment

Bremen

CG VR

Thank you!