CAMPVis – A Game Engine-inspired Research Framework for Medical Imaging and Visualization

Christian Schulte zu Berge[1], Artur Grunau[1], Hossain Mahmud[1], and Nassir Navab[1,2]

[1] Computer Aided Medical Procedures, Technische Universität München, Munich, Germany
[2] Computer Aided Medical Procedures, Johns Hopkins University, Baltimore, MD, USA

Overview

Clinical Motivation

- Intra-operative visualization provides surgeons with powerful support
- A broad range of imaging modalities available, each with different strengths and weaknesses
- Deployment in operating room yields various challenges for software developers

Approach

- 3D video games simulate complex environments providing real-time visualization
- MMOGs synchronize game state over thousands of computers

→ Use game engine architecture for medical visualization framework

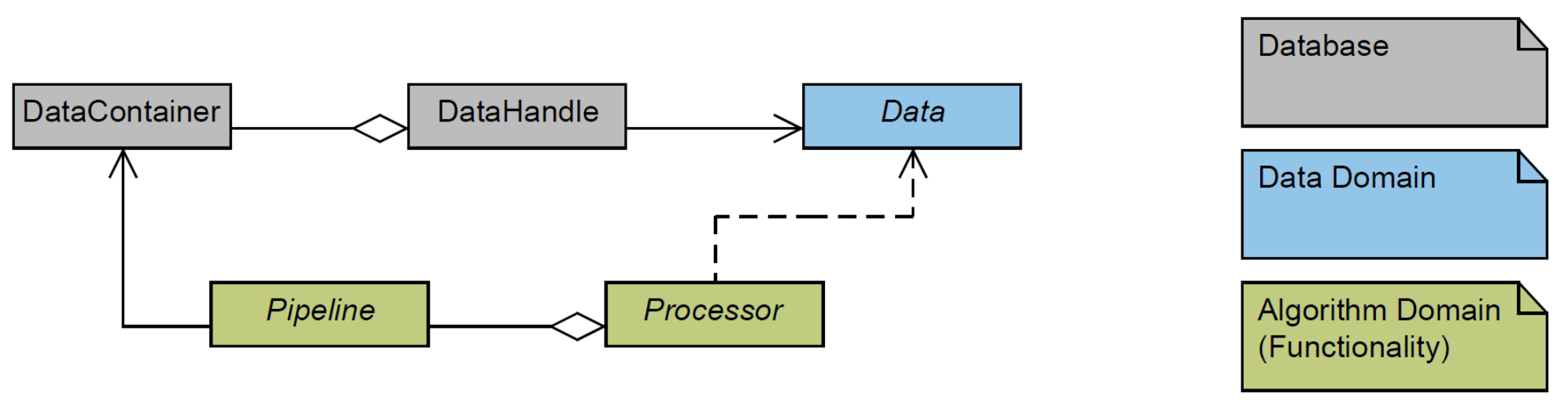
Main Design Goals

- Platform-independent, standard-compliant code base
- Focus on research usage (rapid prototyping) but also allow easy transfer to end-user products
- Sandbox environments for developers
- Support of distributed/decentralized computing

Software Architecture

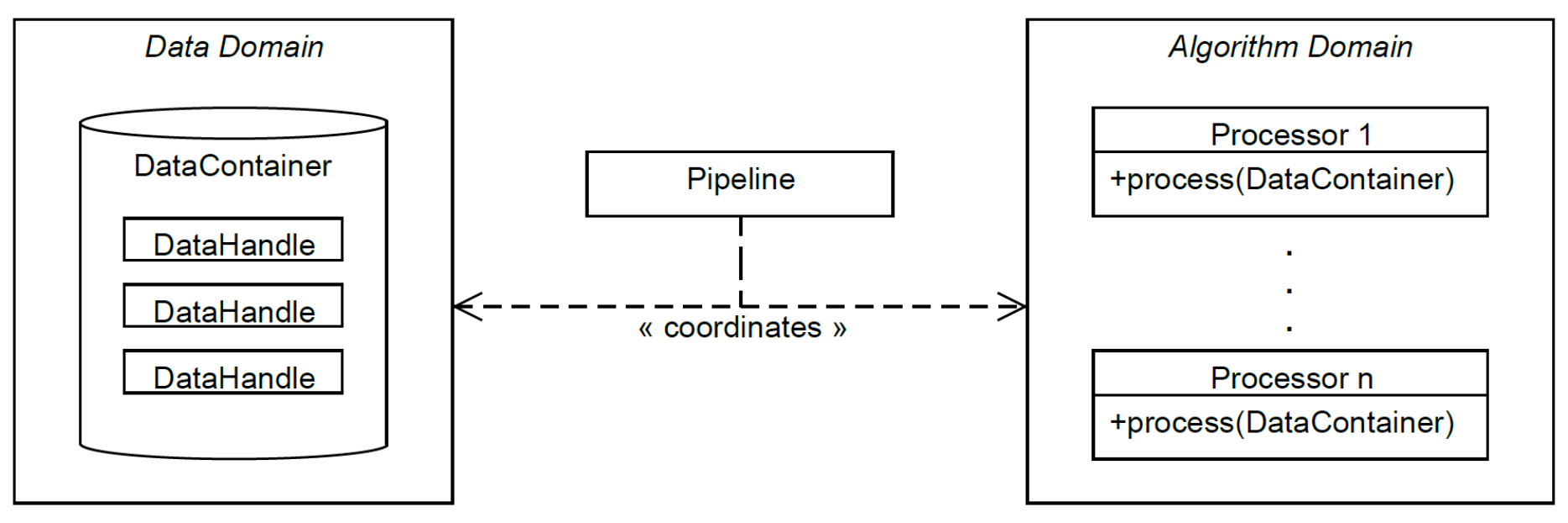
Entity Component System

- Data-driven architecture, which is used often for video game engines [1,2]
- Introduces a database of entities and separates code into components (data domain) and systems (algorithm domain)
- Some modifications to the classic approach for CAMPVis



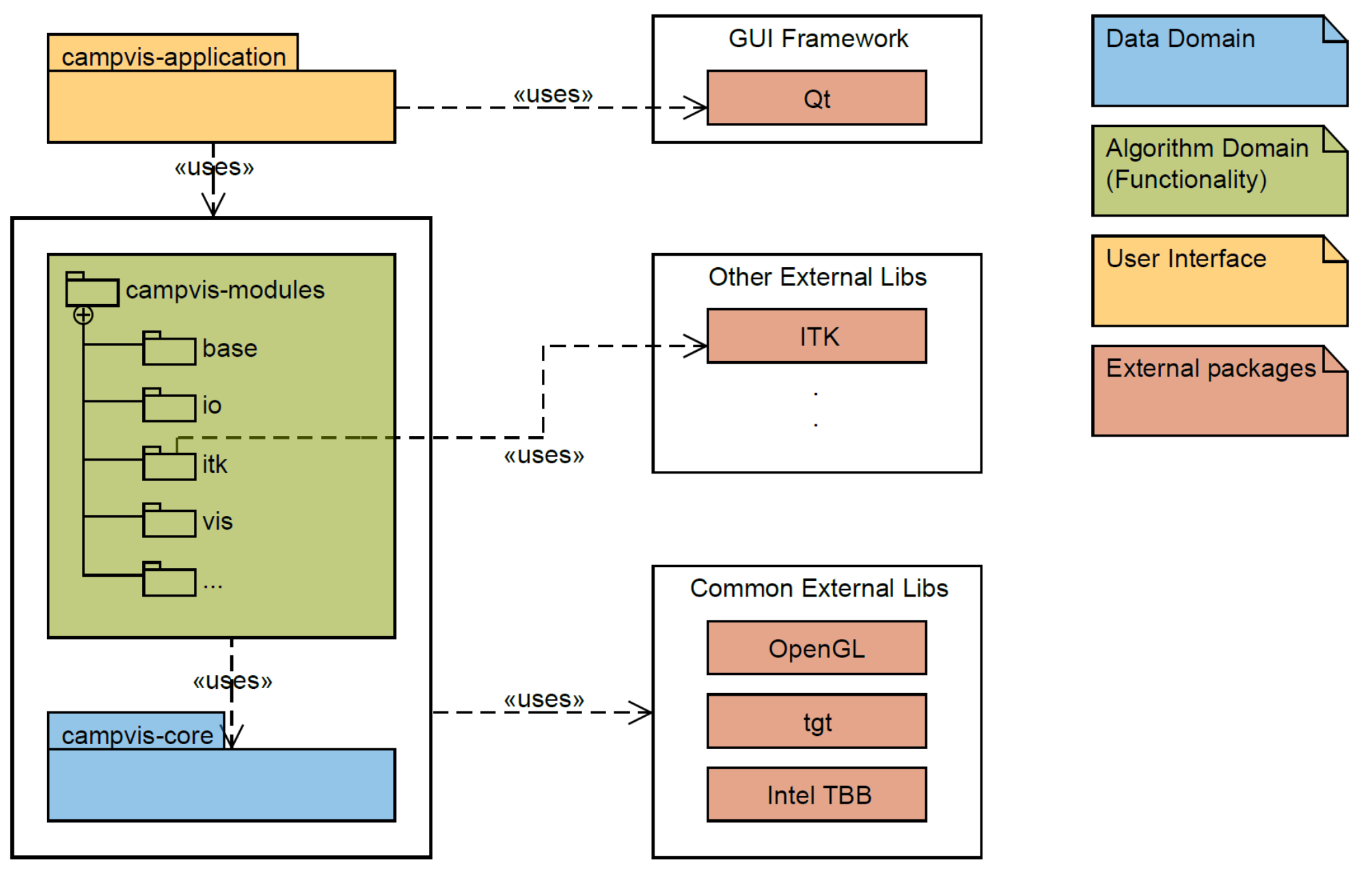
Pipeline concept

- Data stored in DataContainer as database
- Algorithms encapsulated in Processors
- Pipelines coordinate data and algorithms and implement concrete solutions



Module Architecture

- Separation of data and algorithm domain
- Separate GUI package
- Provides development-sandboxes



Selected Framework Features

Property System

Allows for configuration of algorithms

- Implicit getter/setter methods
- Ensures thread safety
- Automatically generated GUI

Network Communication

CAMPCom [3] and OpenIGTLink [4] support

- Real-time streaming of images, tracking information, control commands, etc.

Signal Manager

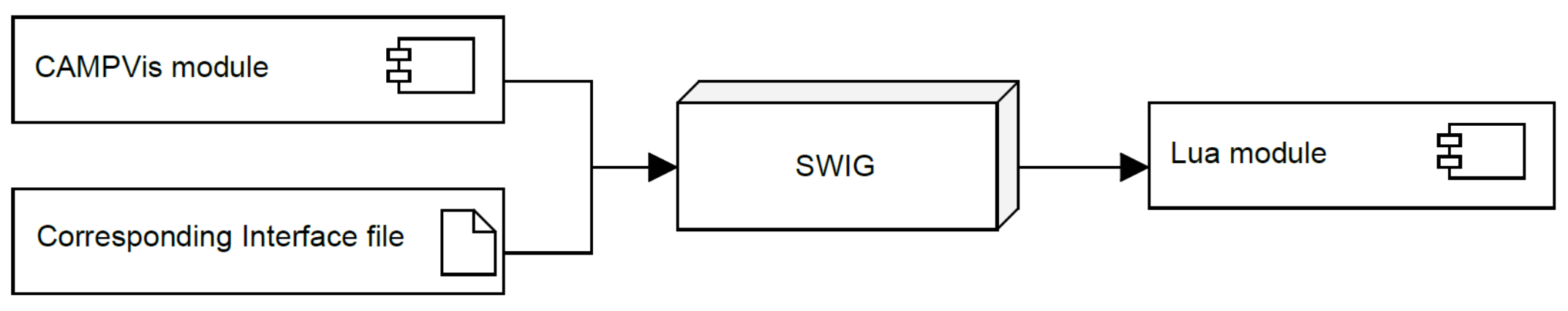
Fusion of event manager messaging concept [2] with signal-slot pattern

- Signal processing runs its own thread
- Direct connection between sender and receiver → very flexible
- All messaging performed through central manager → easy tracking and monitoring
- Both synchronous and asynchronous messaging supported

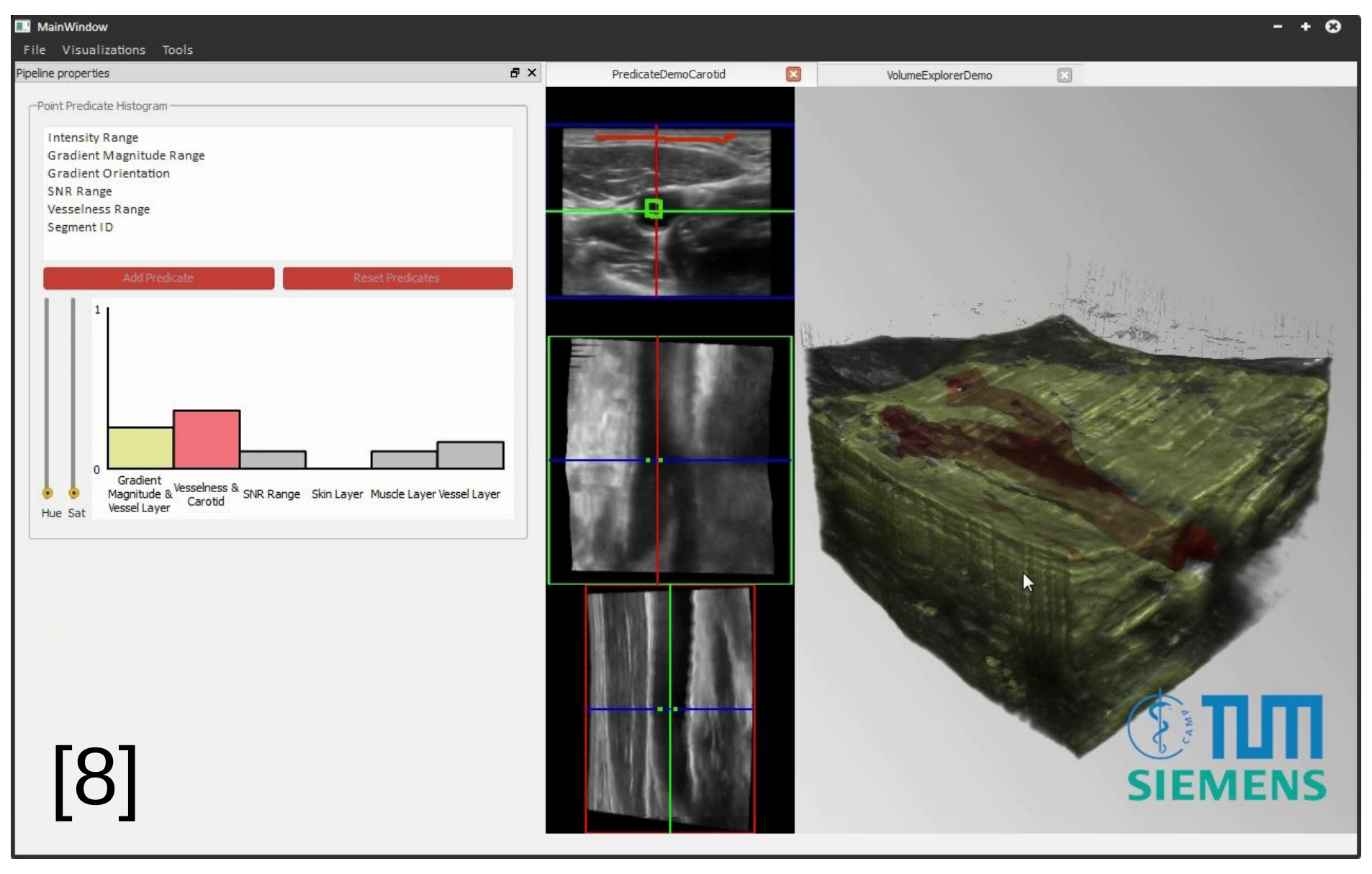
Scripting Layer

Optional Lua [5] scripting layer available

- Scripting console to inspect and modify data model at runtime
- Algorithms and pipelines can be implemented as Lua script instead of C++
- Semi-automated Lua binding generation through SWIG [6]

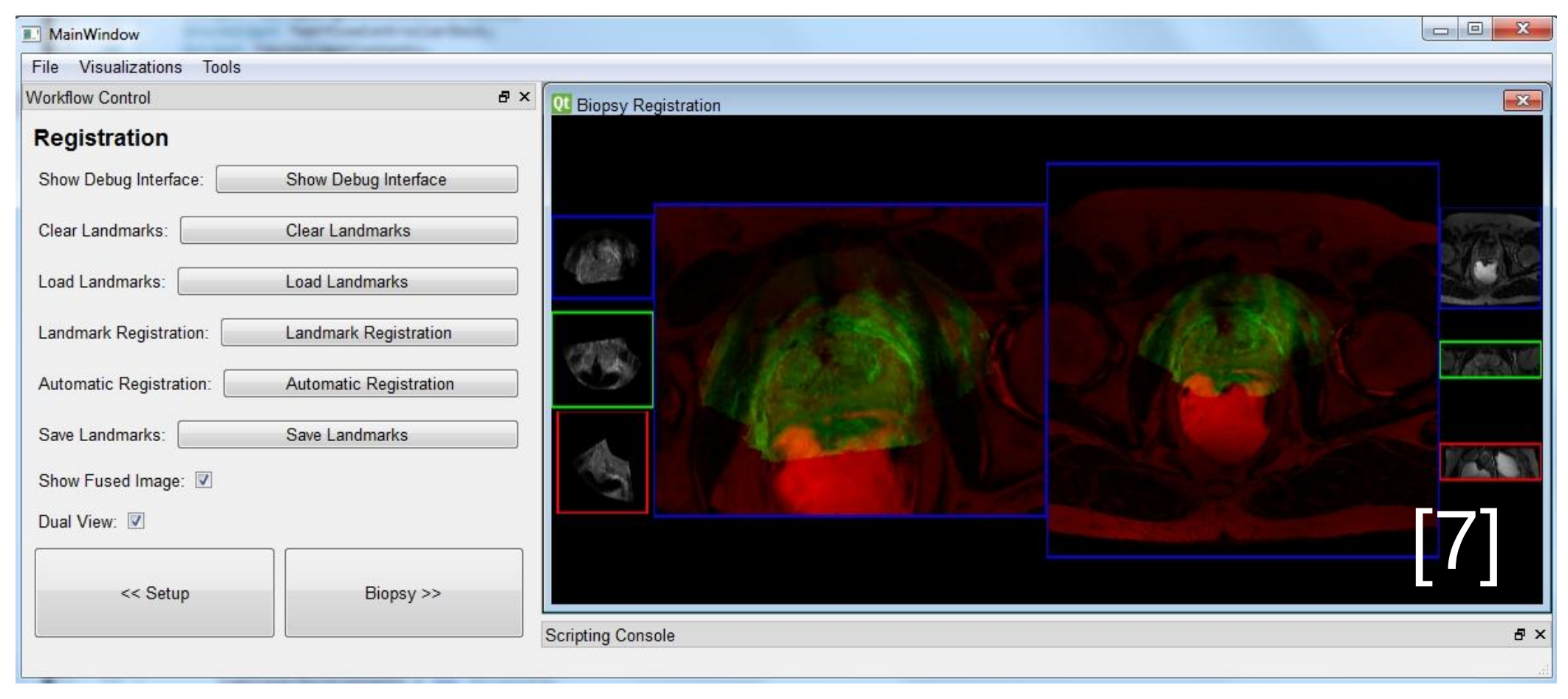


Applications


[8]

So far, two large projects were successfully realized using the CAMPVis software framework:

- Multi-modal image-guided prostate biopsy framework [7] in collaboration with Klinikum Rechts der Isar, Munich, Germany
- Reference implementation of a predicate-based focus-and-context volume rendering technique for 3D ultrasound [8]


[7]

References

[1] A. Martin. Entity Systems are the future of MMOG development (2007)
[2] M. McShary. Game Coding Complete Fourth Edition (2013)
[3] A. Schoch et al. A Lightweight and Portable Communication Framework for Multimodal Image-Guided Therapy (2013)
[4] J. Tokuda et al. OpenIGTLink: an open network protocol for image-guided therapy environment (2009)
[5] W. Celes and L. H. de Figueiredo. Lua (programming language). `http://www.lua.org/`
[6] D. Beazley. Simplified Wrapper and Interface Generator (SWIG). `http://swig.org/`
[7] A. Shah, et al. An open source multimodal image-guided prostate biopsy framework (2014)
[8] C. Schulte zu Berge et al. Predicate-based focus-and-context visualization for 3D ultrasound (2014)

SIEMENS
software campus
Bundesministerium für Bildung und Forschung
MICCAI 15 MUNICH