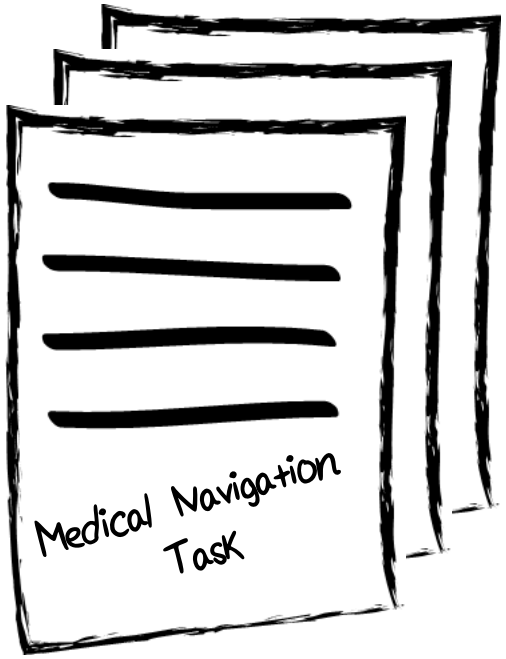
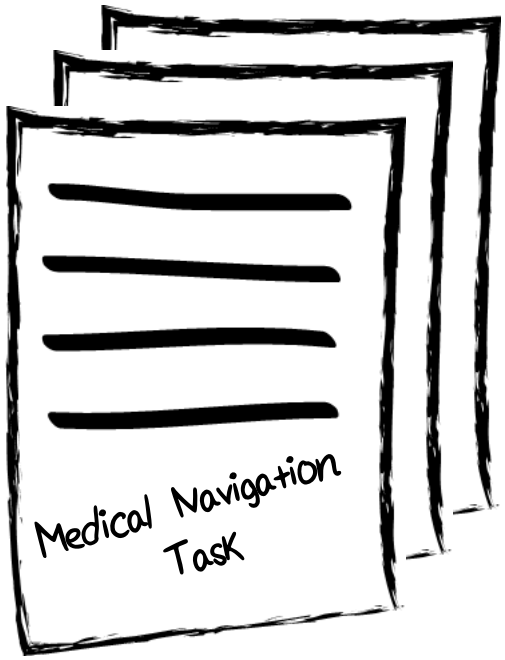
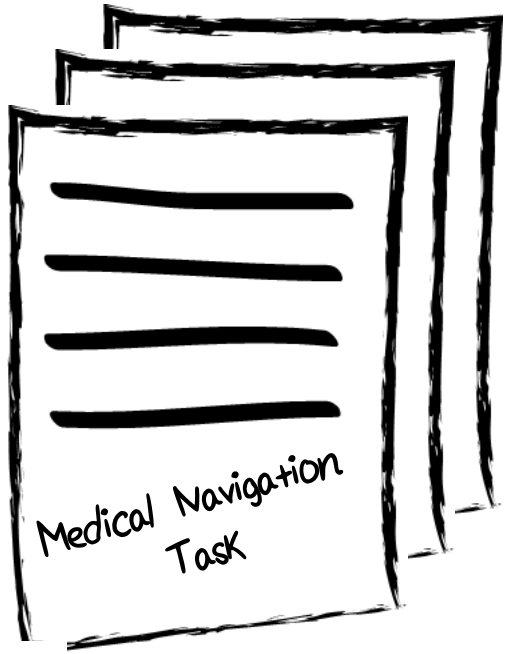

Medical Navigation
Task

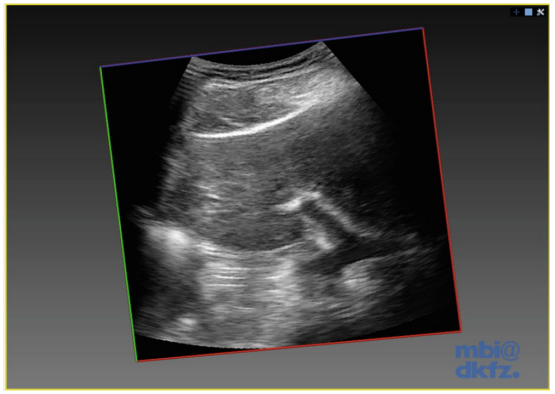

Medical Navigation
Task

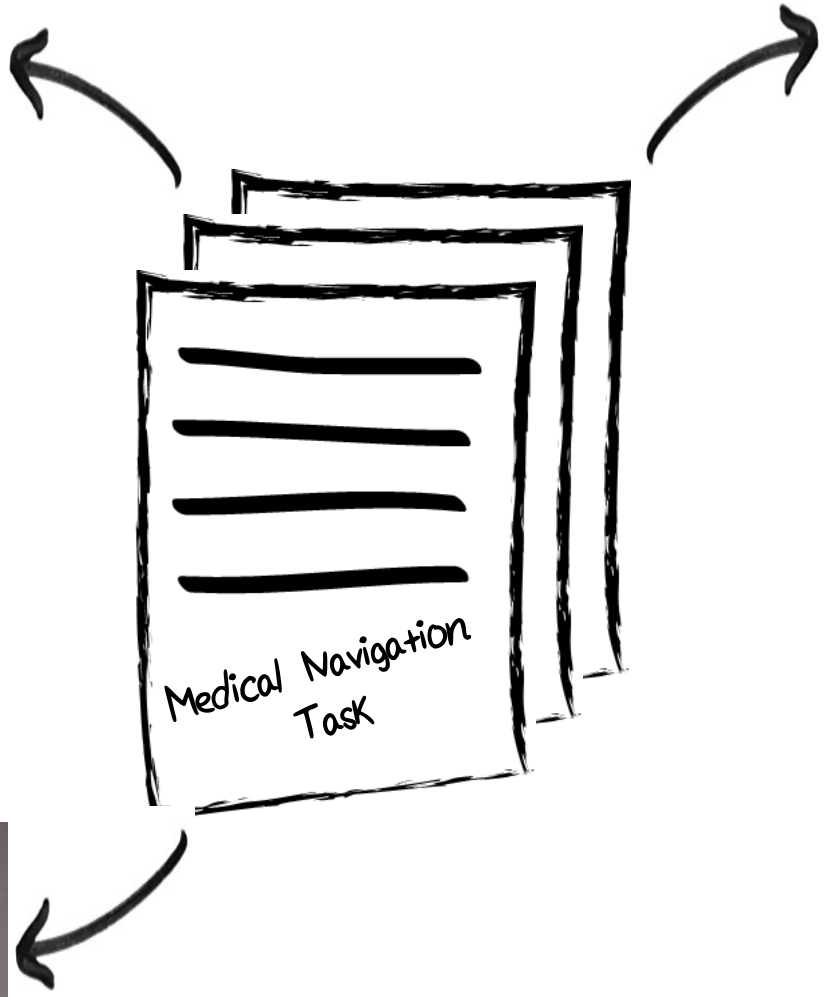

ZONARE


Medical Navigation
Task


ZONARE

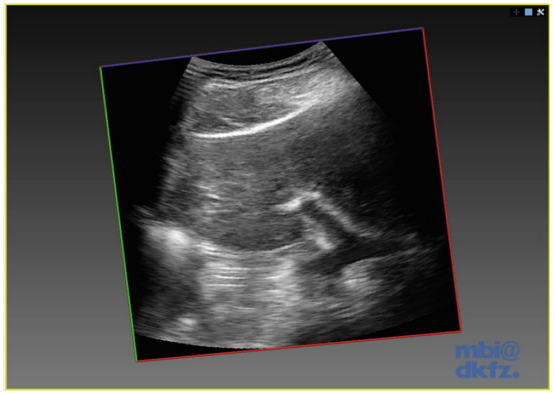

AURORA
NDI


mbi@
dkfz.

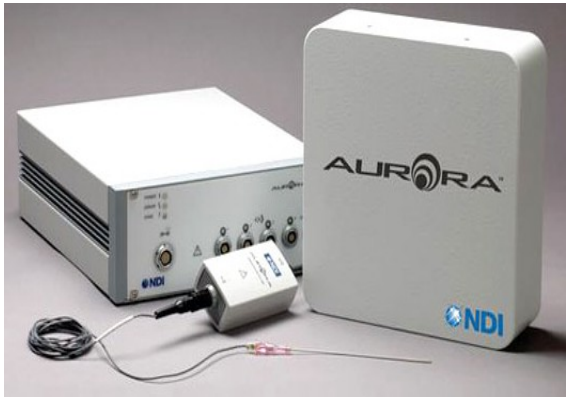

Medical Navigation
Task


ZONARE

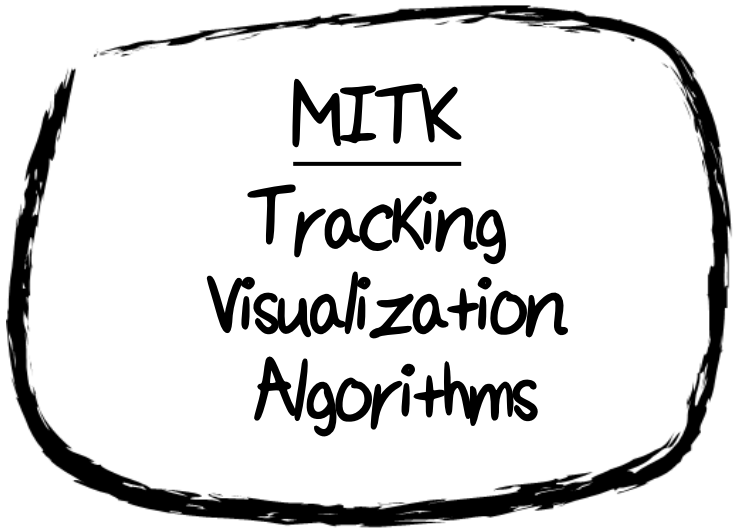

AURORA
NDI
NDI

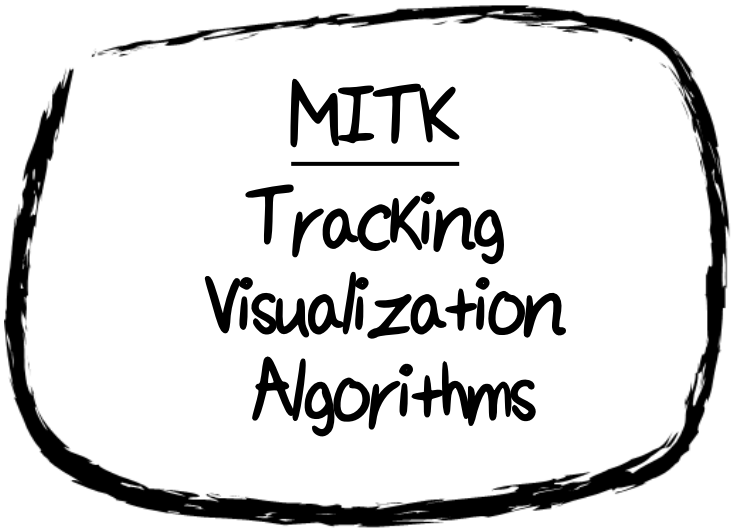


Medical Navigation Task

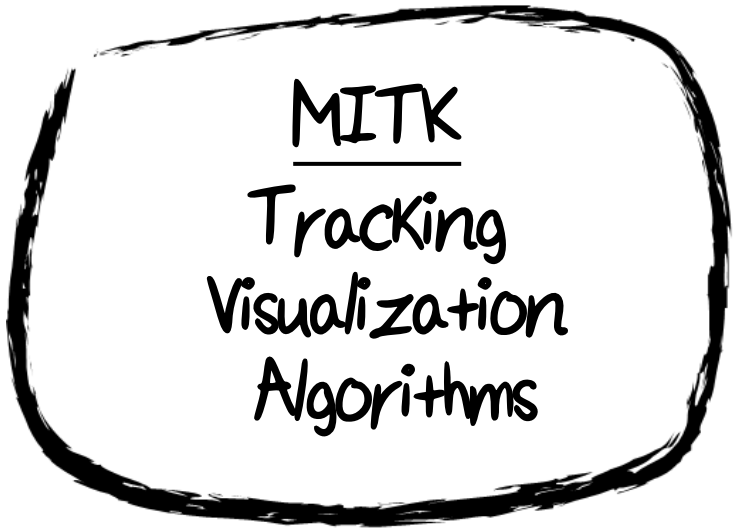


Algorithms/ Calibration


MITK
Tracking
Visualization
Algorithms



US_Device is not supported
Some Algorithms are missing



US_Device is not supported
Some Algorithms are missing

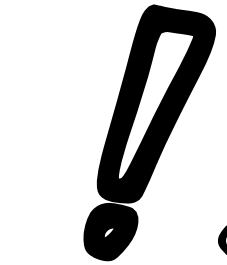
MITK

Tracking
Visualization
Algorithms

US_Device is not supported
Some Algorithms are missing

MITK

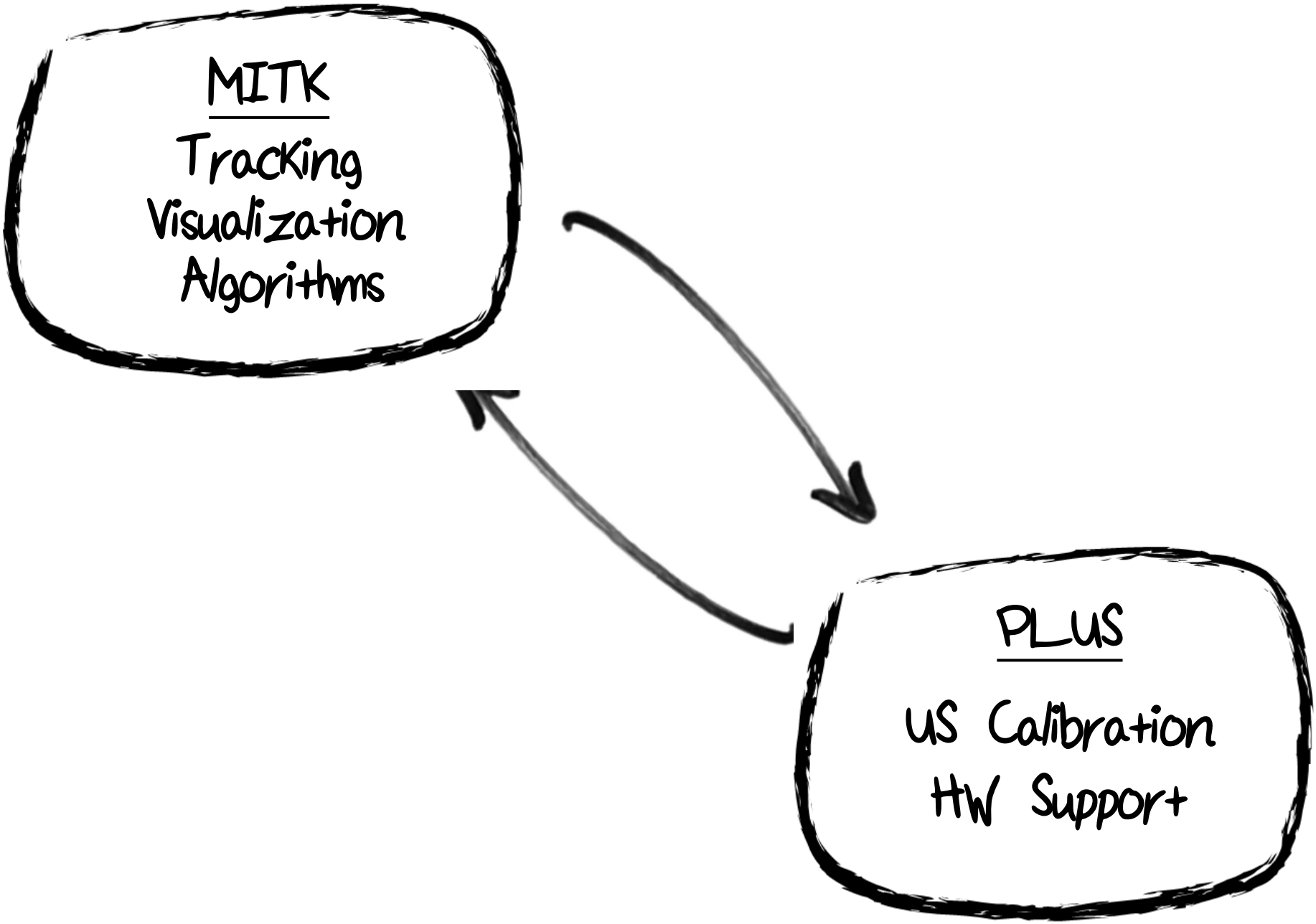
Tracking
Visualization
Algorithms

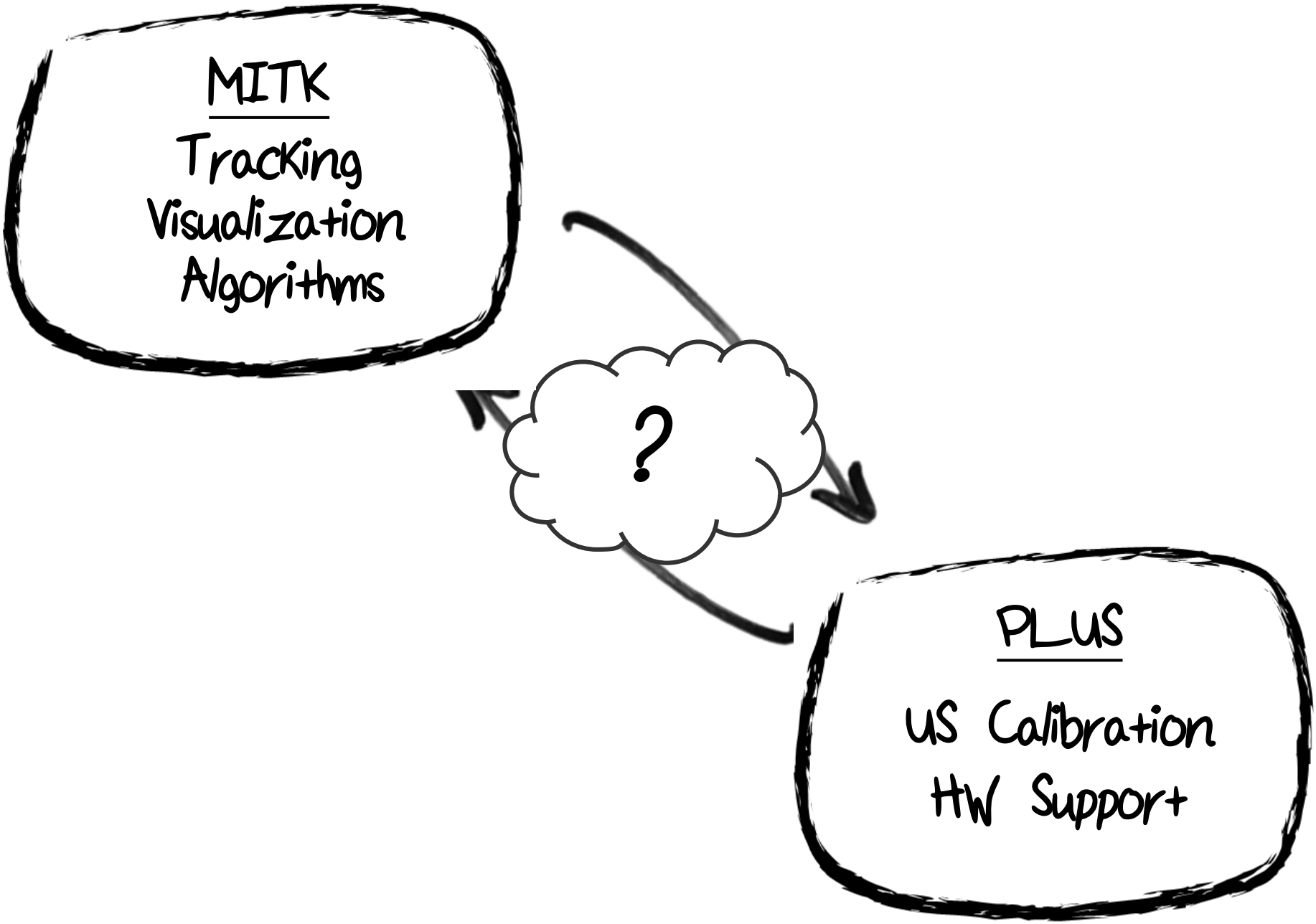


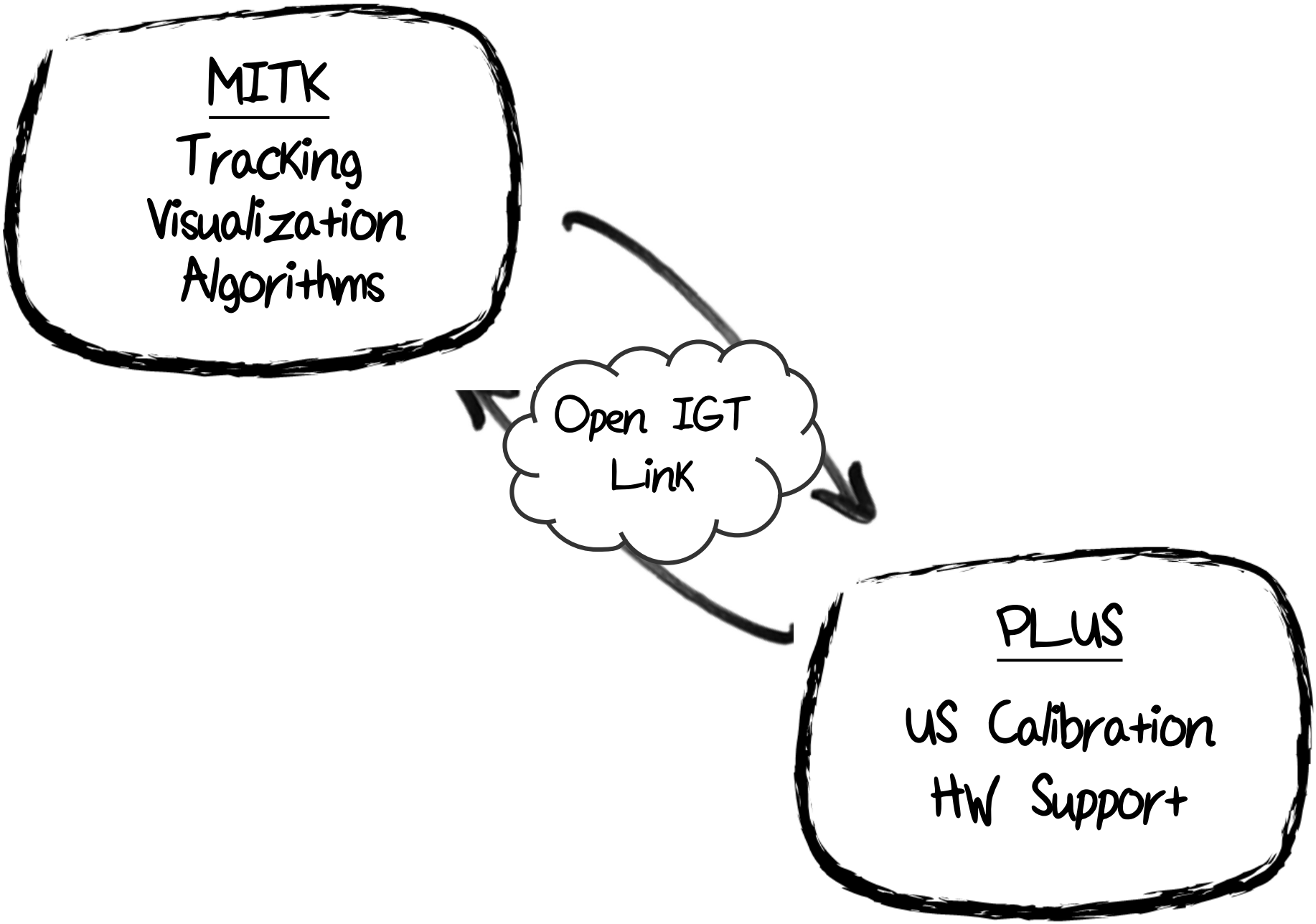
US_Device is not supported
Some Algorithms are missing

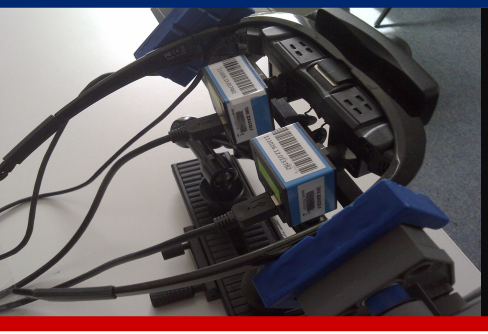
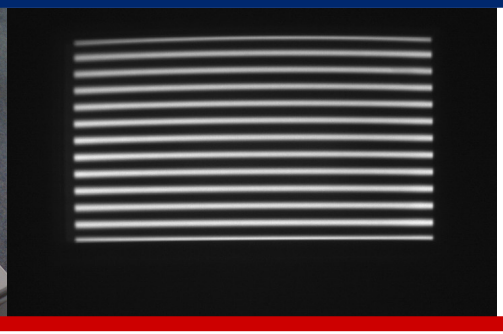
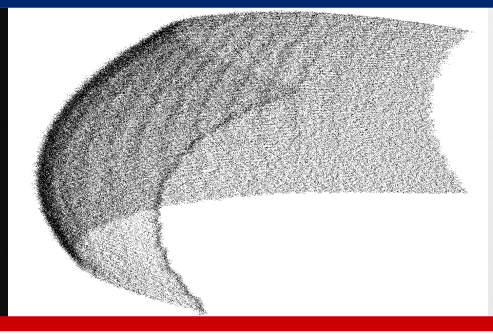
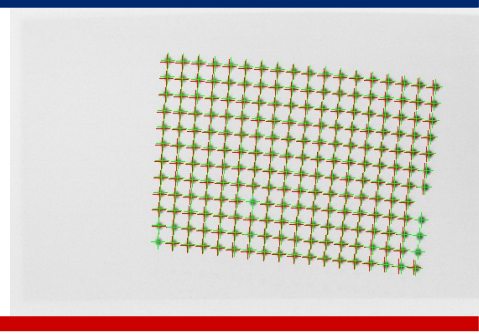
PLUS

US Calibration
HW Support


MITK
Tracking
Visualization
Algorithms
PLUS
US Calibration
HW Support


MITK
Tracking
Visualization
Algorithms
?
PLUS
US Calibration
HW Support


MITK
Tracking
Visualization
Algorithms
Open IGT
Link
PLUS
US Calibration
HW Support

OpenIGTLink Support for MITK

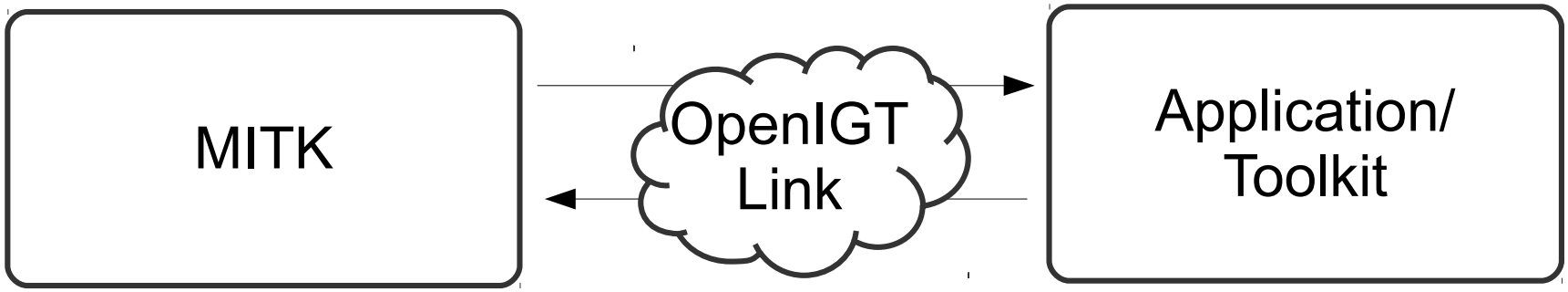
MITK Users Day 2015

Martin Klemm, Lab. for Computer-assisted Medicine, University of Applied Sciences Offenburg, Germany
Alfred Franz, Junior Group Computer-assisted Interventions , DKFZ, Heidelberg, Germany
27.04.2015

Hochschule Offenburg
University of Applied Sciences
E+I
Elektrotechnik und Informationstechnik
dkfz.
DEUTSCHES KREBSFORSCHUNGSZENTRUM
IN DER HELMHOLTZ-GEMEINSCHAFT

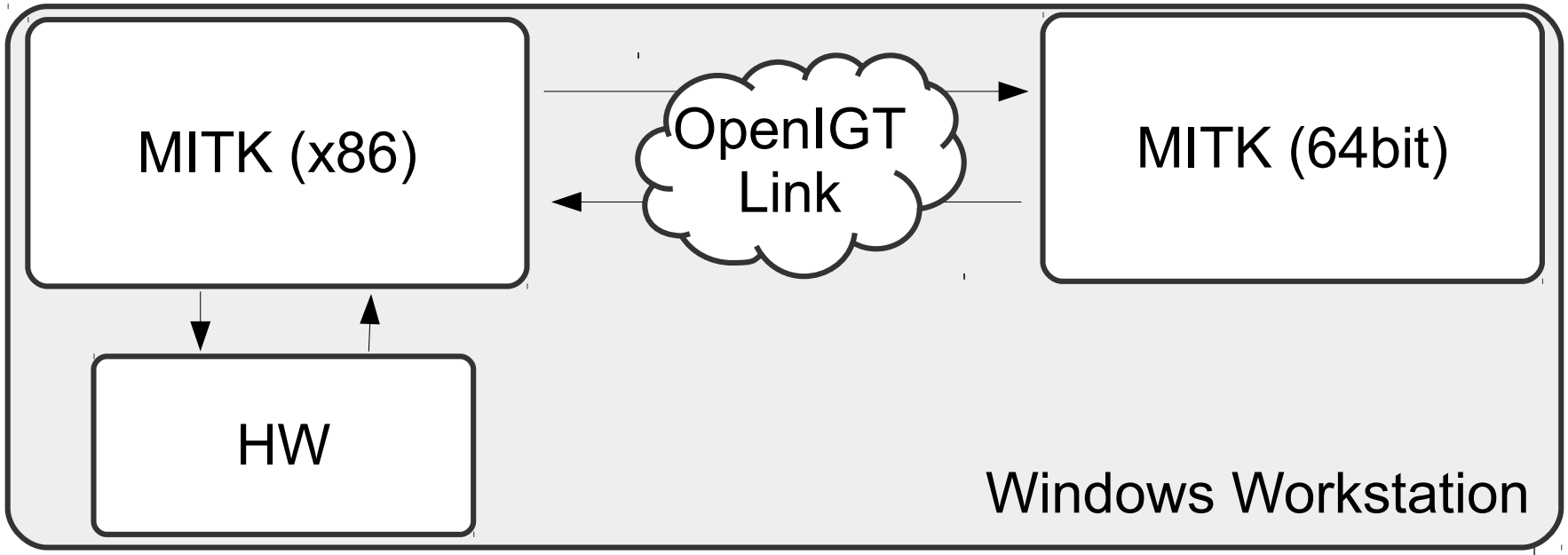
Motivation

- Interoperate with other applications or toolkits because of additional functionality
- Examples: 3DSlicer, PLUS, MUSiiC and many more



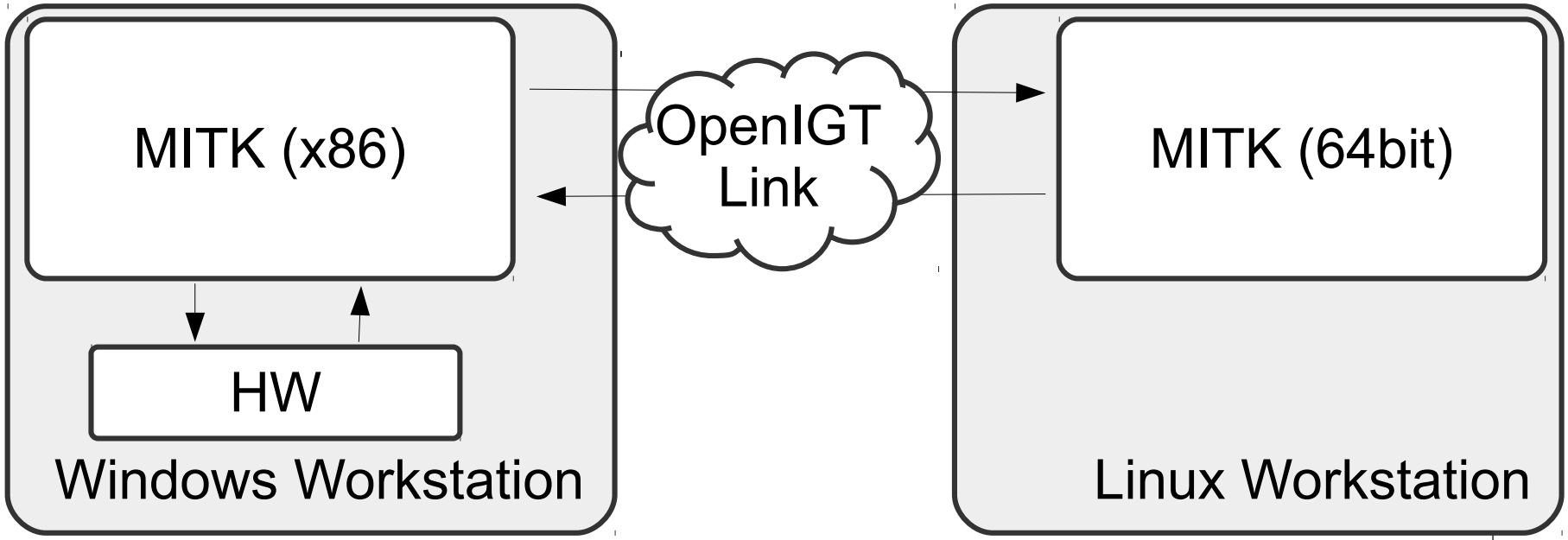
Motivation

- Interoperate with other MITK instance with other build type (x86/64bit) or on other operating system (Linux/Windows/OSX)
- Examples:
  - MITK is compiled for 64bit and HW driver are only available for 32bit



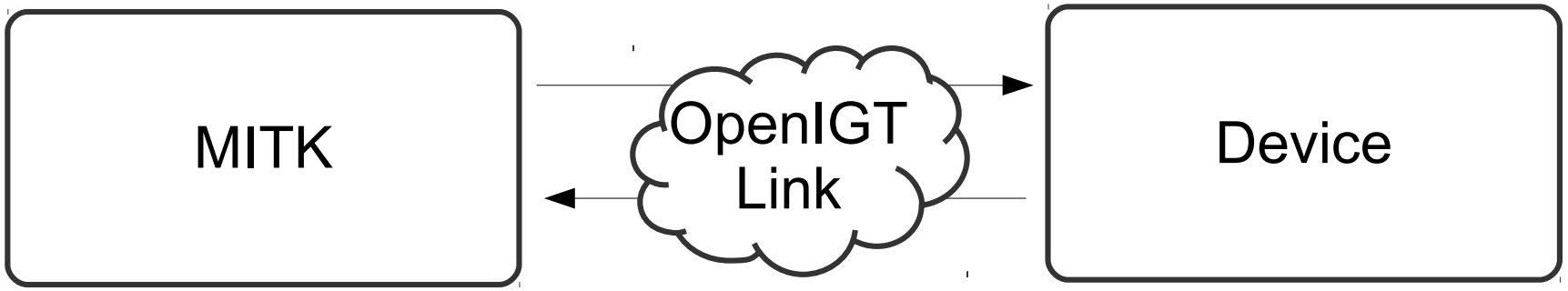
Motivation

- Interoperate with other MITK instance with other build type (x86/64bit) or on other operating system (Linux/Windows/OSX)
- Examples:
  - MITK is compiled for 64bit and HW driver are only available for 32bit
  - MITK runs on Linux workstation and HW driver is only available for Windows



Motivation

- Use hardware that is able to speak OpenIGTLink natively
- Example: US-Device by Verasonics



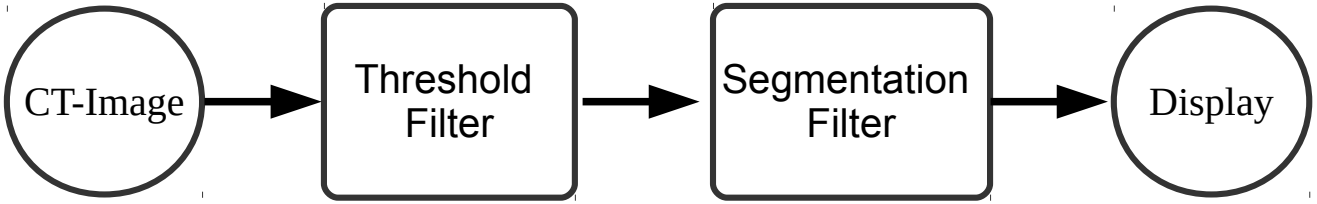
Open Image Guided Therapy Link

- Open-Source network protocol
- Originally developed for IGT environments
- De facto standard in medical applications
- Integrated into several toolkits: 3D Slicer, PLUS, IGSTK, MUSiiC, MeVisLab
- Runs in Application Layer on top of TCP (or UDP)
- Predefined types cover most applications
- Extensible for custom types
- Support for data queries

MITK - Pipelines

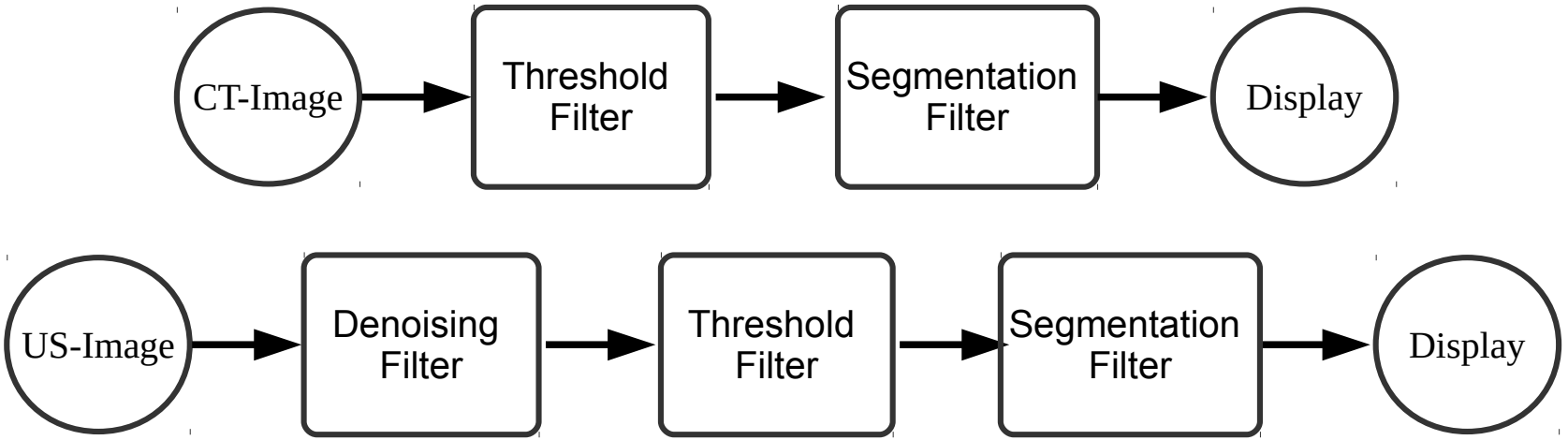
[1] http://docs.mitk.org/nightly/PipelineingConceptPage.html

MITK - Pipelines



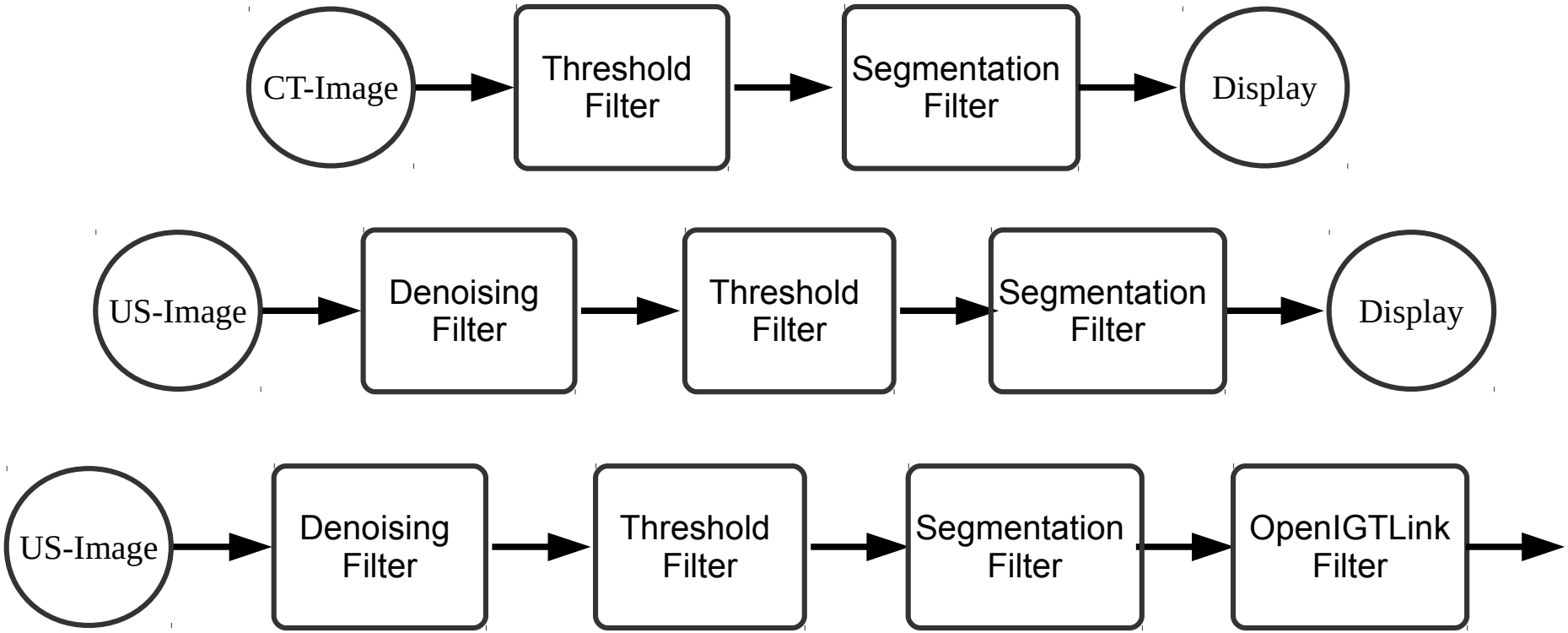
[1] http://docs.mitk.org/nightly/PipelineingConceptPage.html

MITK - Pipelines



[1] http://docs.mitk.org/nightly/PipelineingConceptPage.html

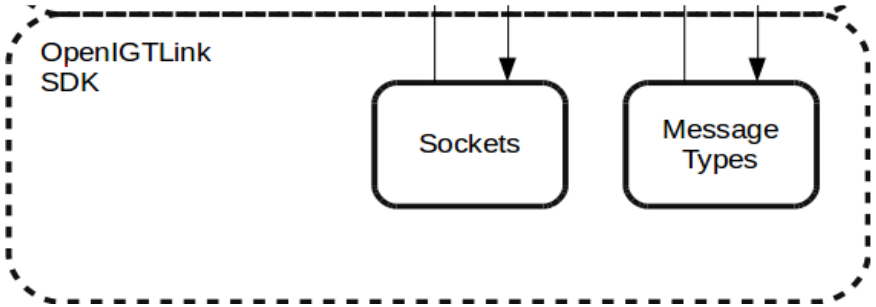
MITK - Pipelines



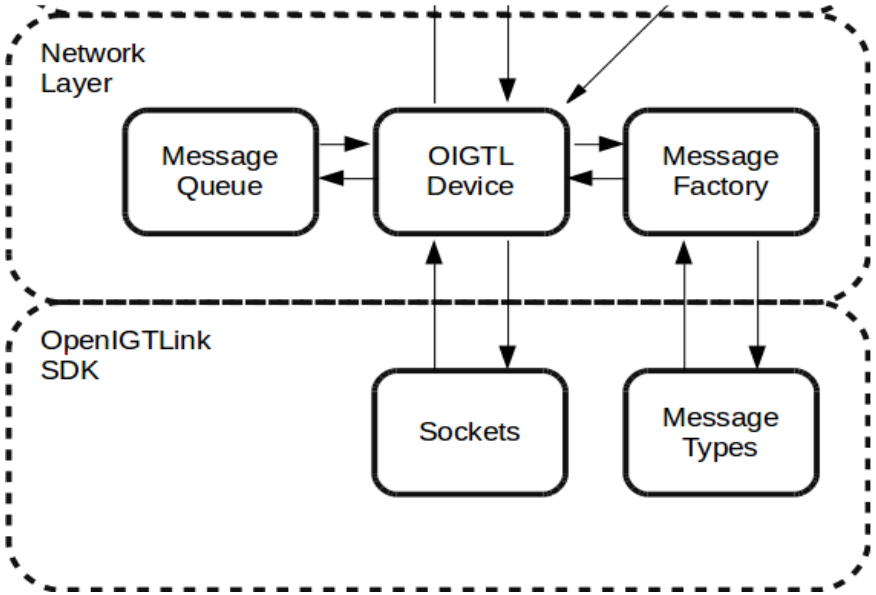
[1] http://docs.mitk.org/nightly/PipelineingConceptPage.html

Implementation - Overview

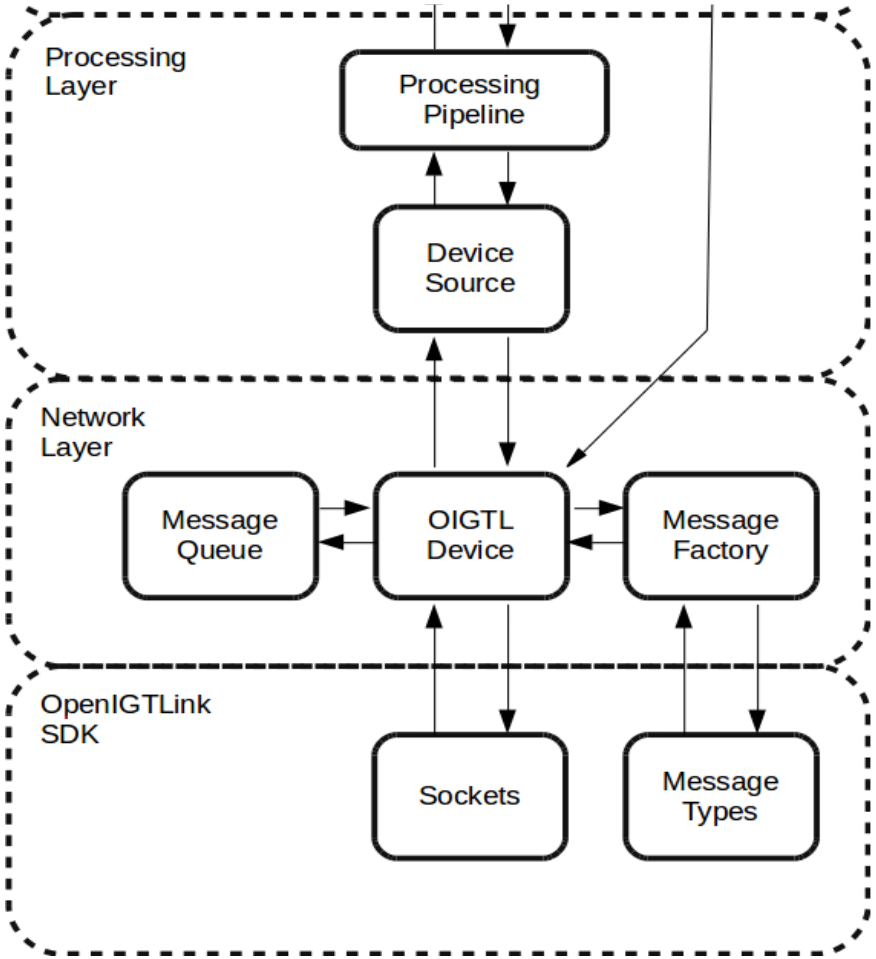
Implementation - Overview



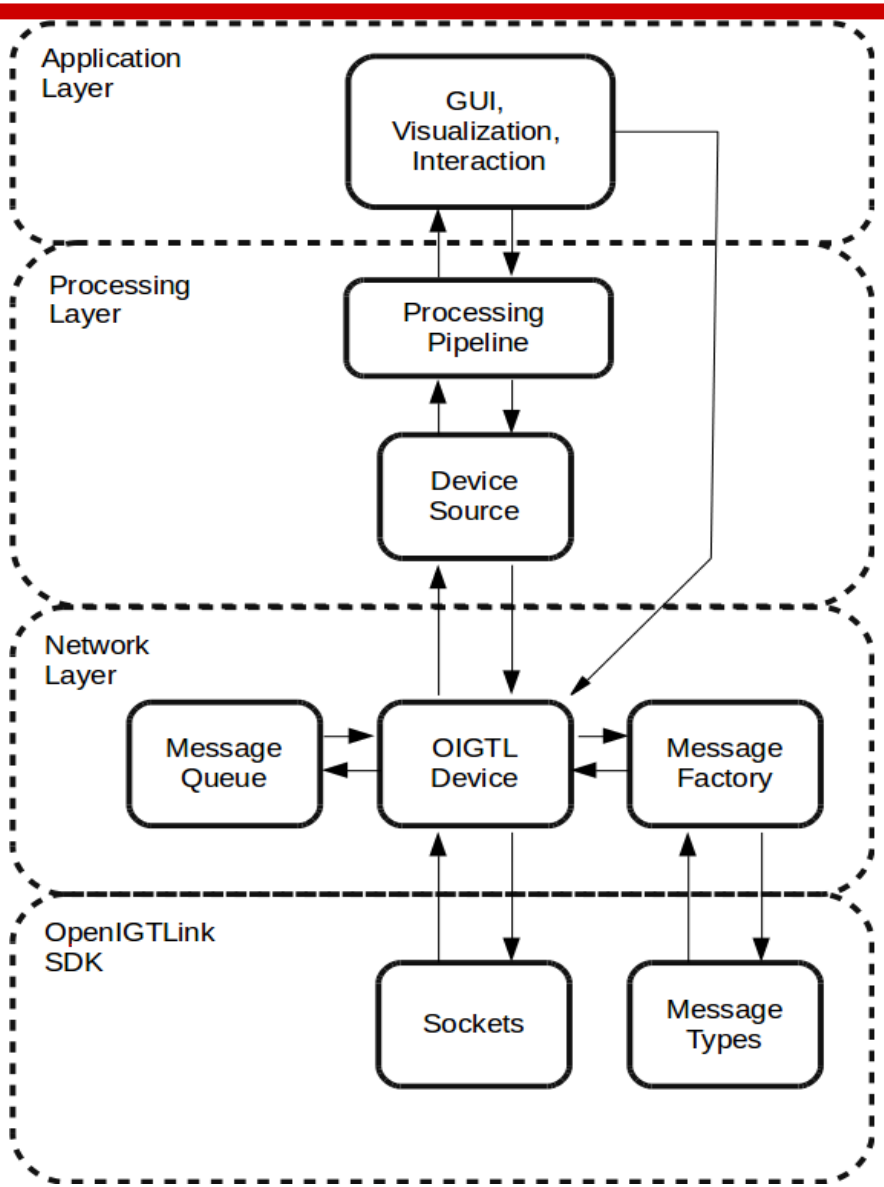
Implementation - Overview



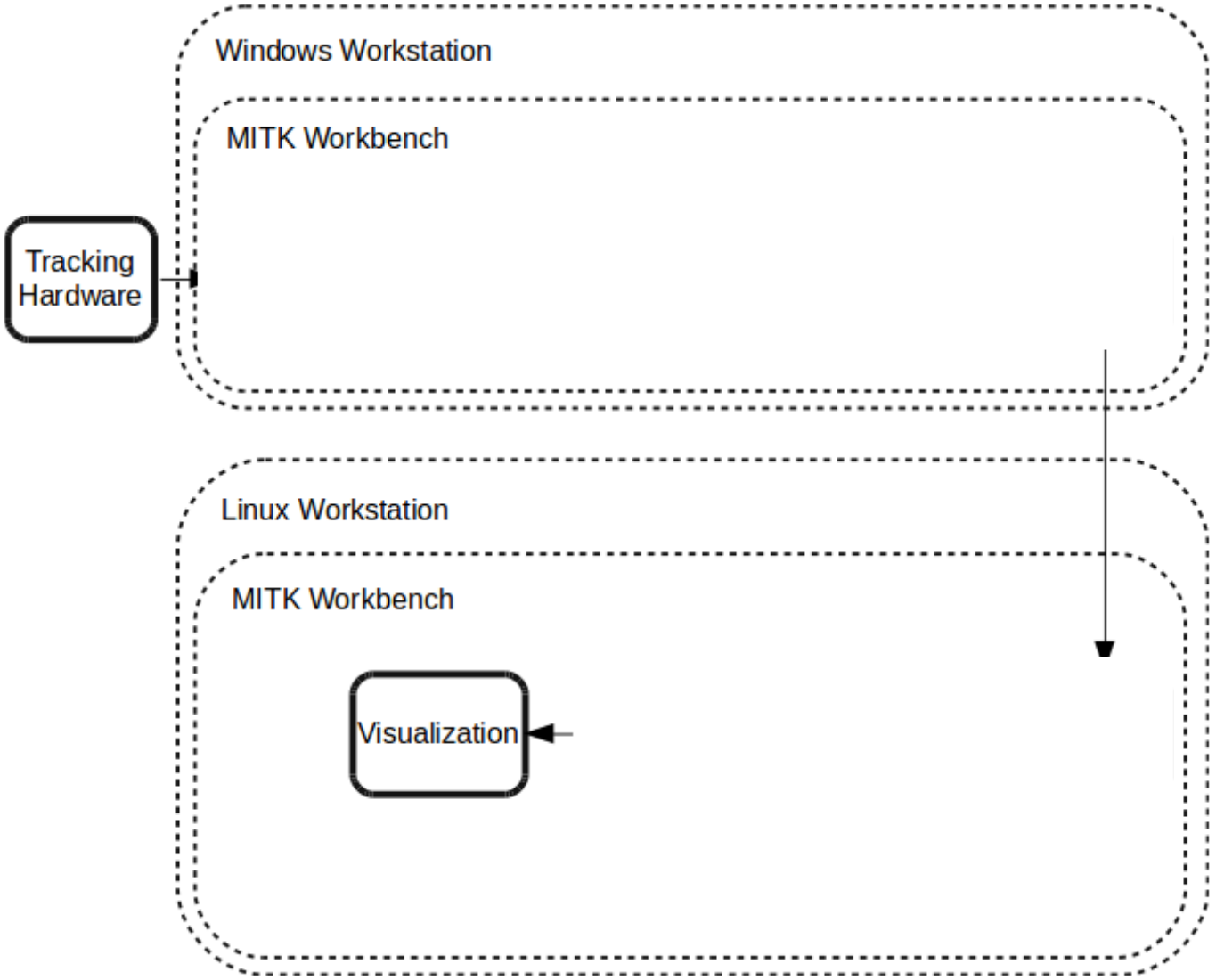
Implementation - Overview



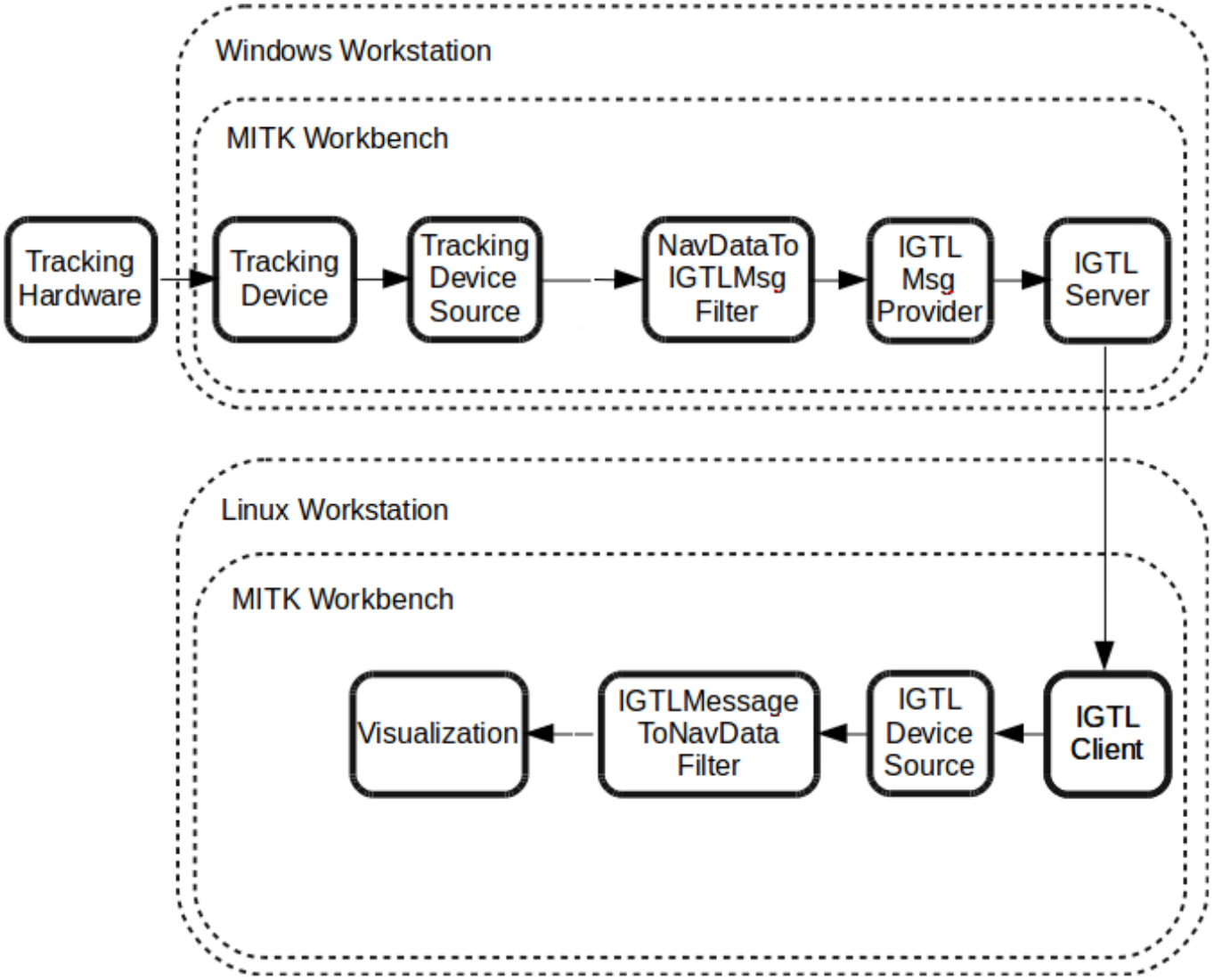
Implementation - Overview



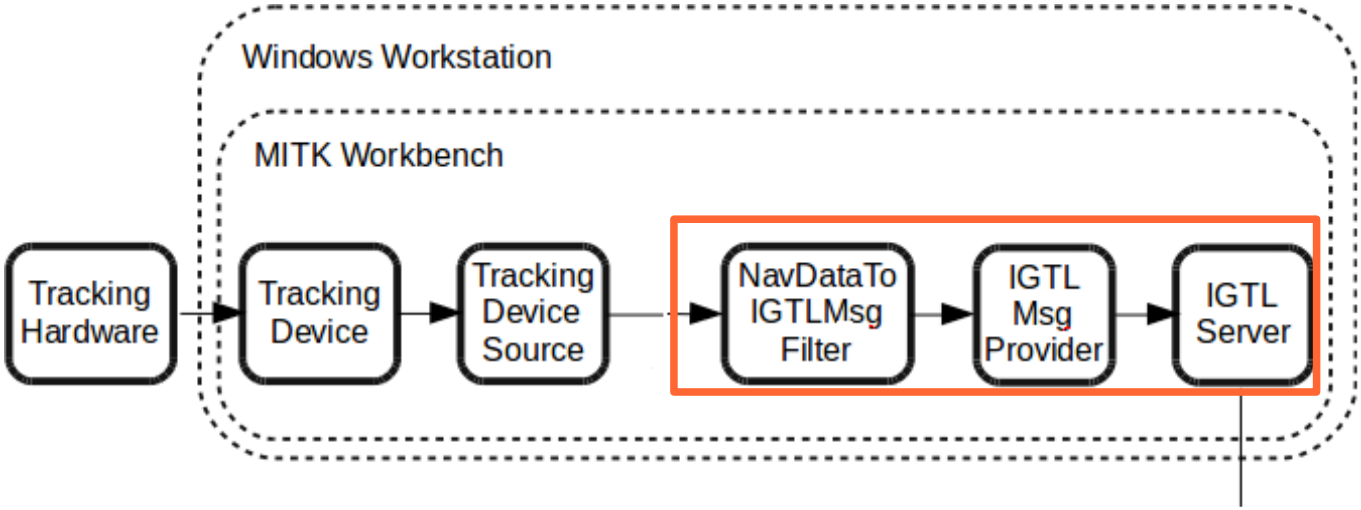
Example


Windows Workstation
MITK Workbench
Tracking Hardware
Linux Workstation
MITK Workbench
Visualization

Example

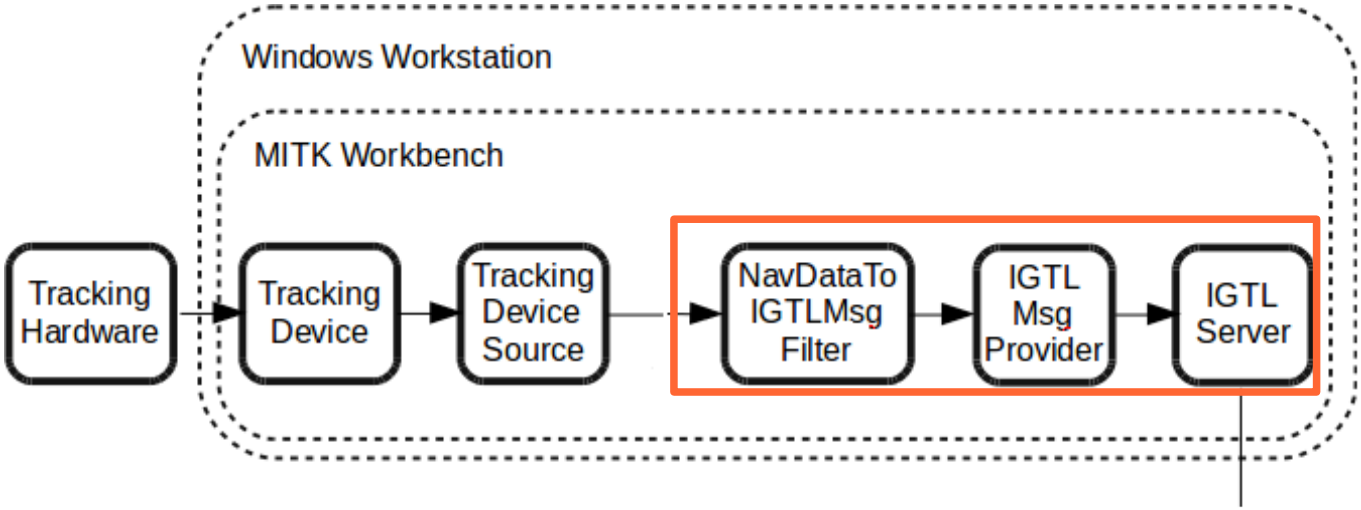

Windows Workstation
MITK Workbench
Tracking Hardware
Tracking Device
Tracking Device Source
NavDataTo IGTLMsg Filter
IGTL Msg Provider
IGTL Server
Linux Workstation
MITK Workbench
Visualization
IGTLMessage ToNavData Filter
IGTL Device Source
IGTL Client

Example




```
//Init tracking device and source, connect, start tracking
conversionFilter->ConnectTo(m_TrackingSource);
conversionFilter->SetOperationMode(TDATA);
conversionFilter->RegisterAsMicroService();

server->SetPortNumber(port);
provider->SetIGTLDevice(server);
provider->Connect();
provider->StartCommunication();
```

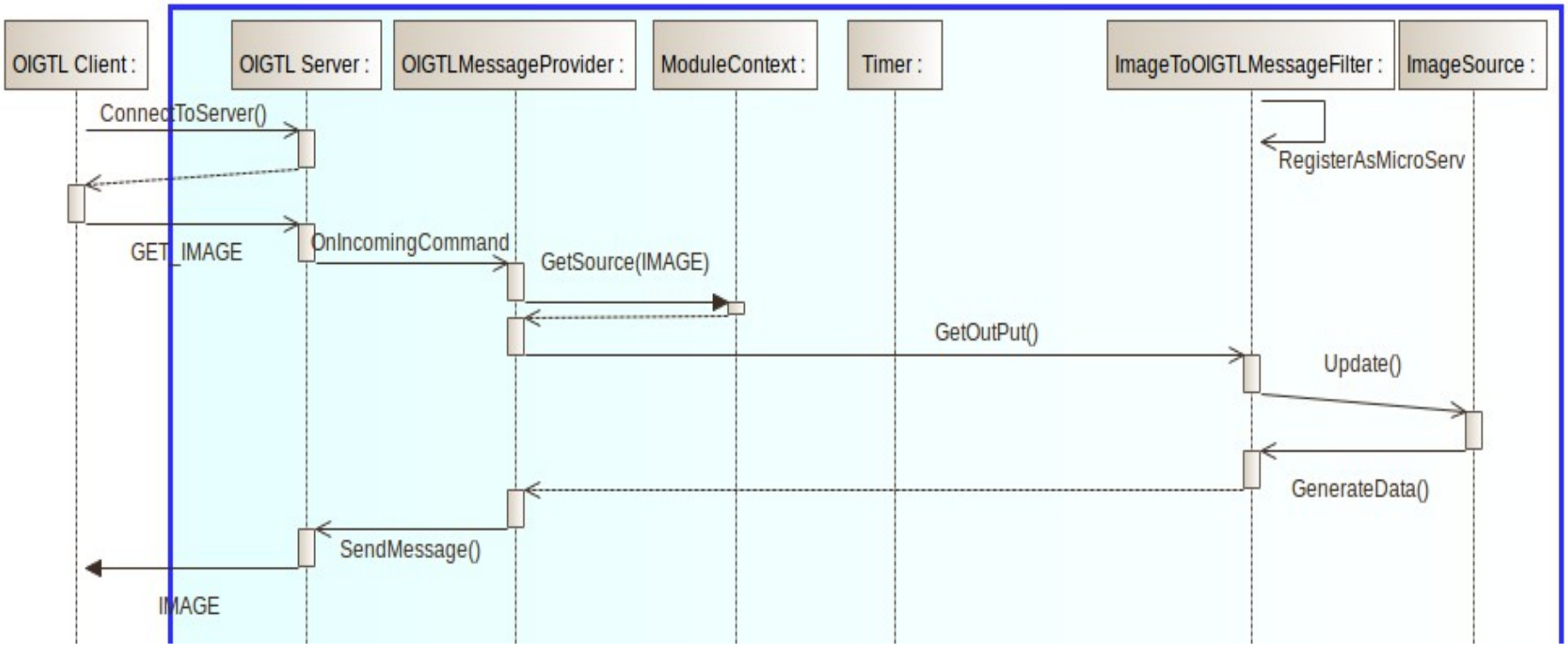
Example




```
//Init tracking device and source, connect, start tracking
conversionFilter->ConnectTo(m_TrackingSource);
conversionFilter->SetOperationMode(TDATA);
conversionFilter->RegisterAsMicroService();

server->SetPortNumber(port);
provider->SetIGTLDevice(server);
provider->Connect();
provider->StartCommunication();
```

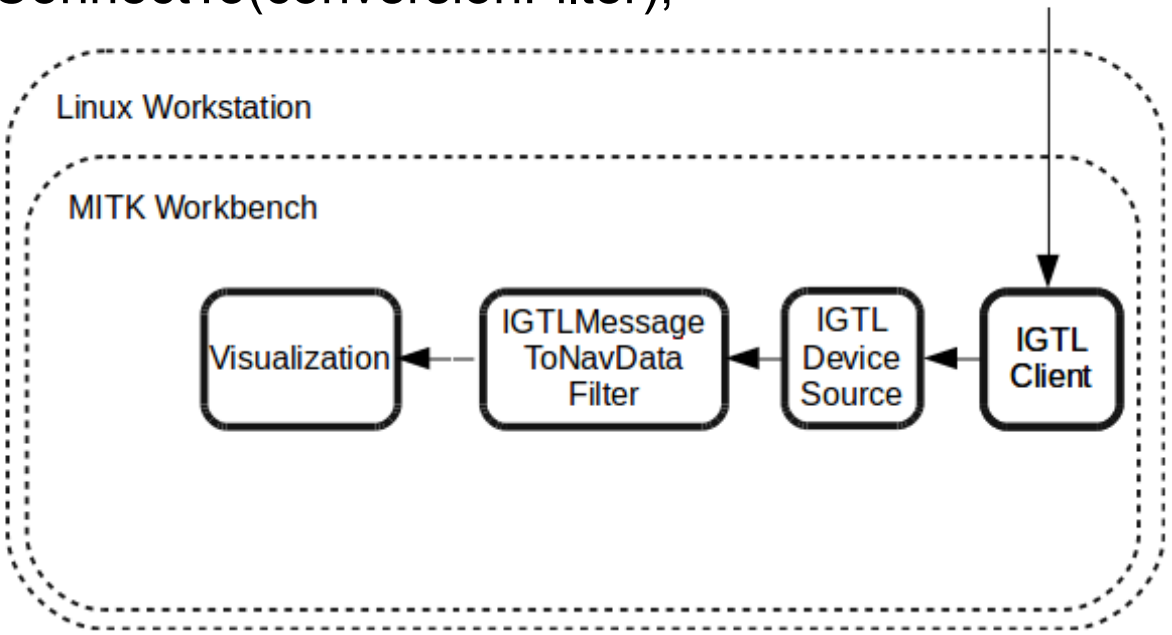
! No connection between provider and conversion filter

Data Queries

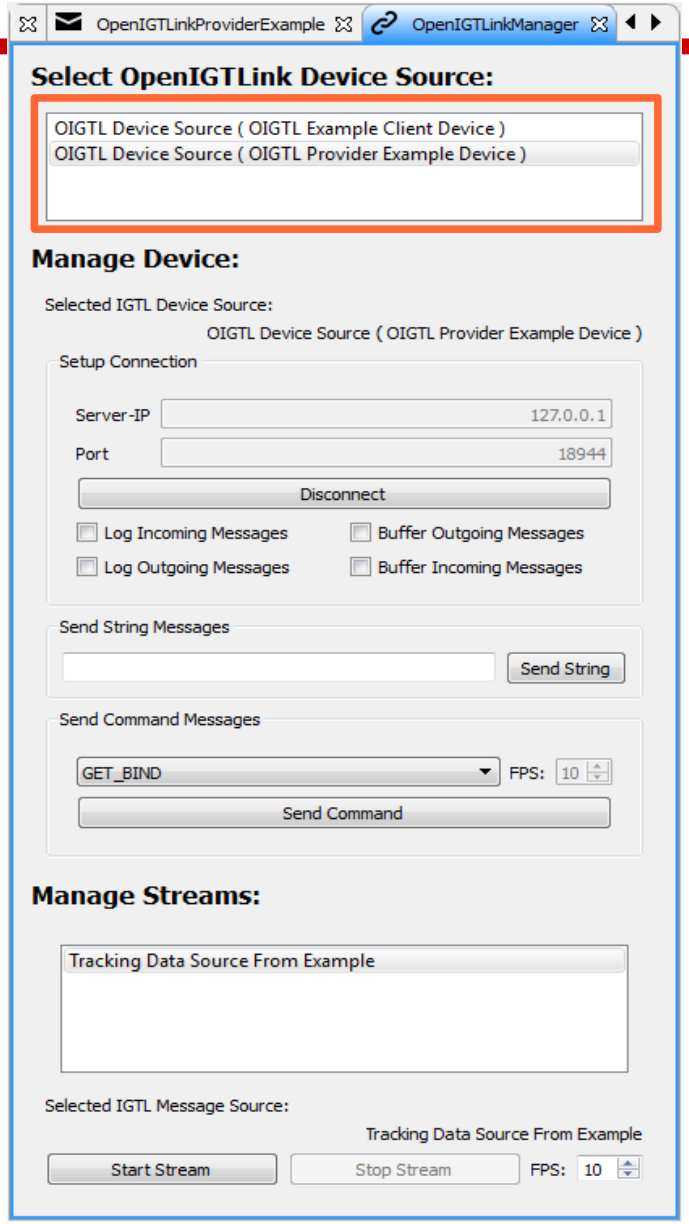


Example

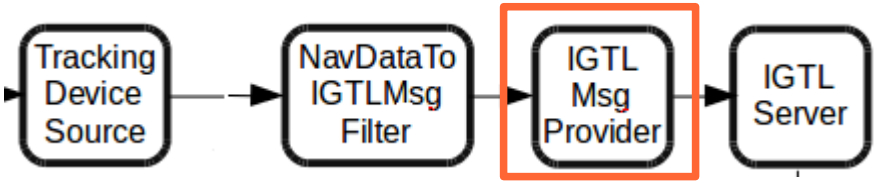
```
//Init all filters and the client
deviceSource->SetIGTLDevice(client);
deviceSource->RegisterAsMicroservice();
client->Connect(hostname, port);
client->SendMessage(STT_TDATA_Message(FPS));

conversionFilter->ConnectTo(deviceSource);
visFilter->ConnectTo(conversionFilter);
```



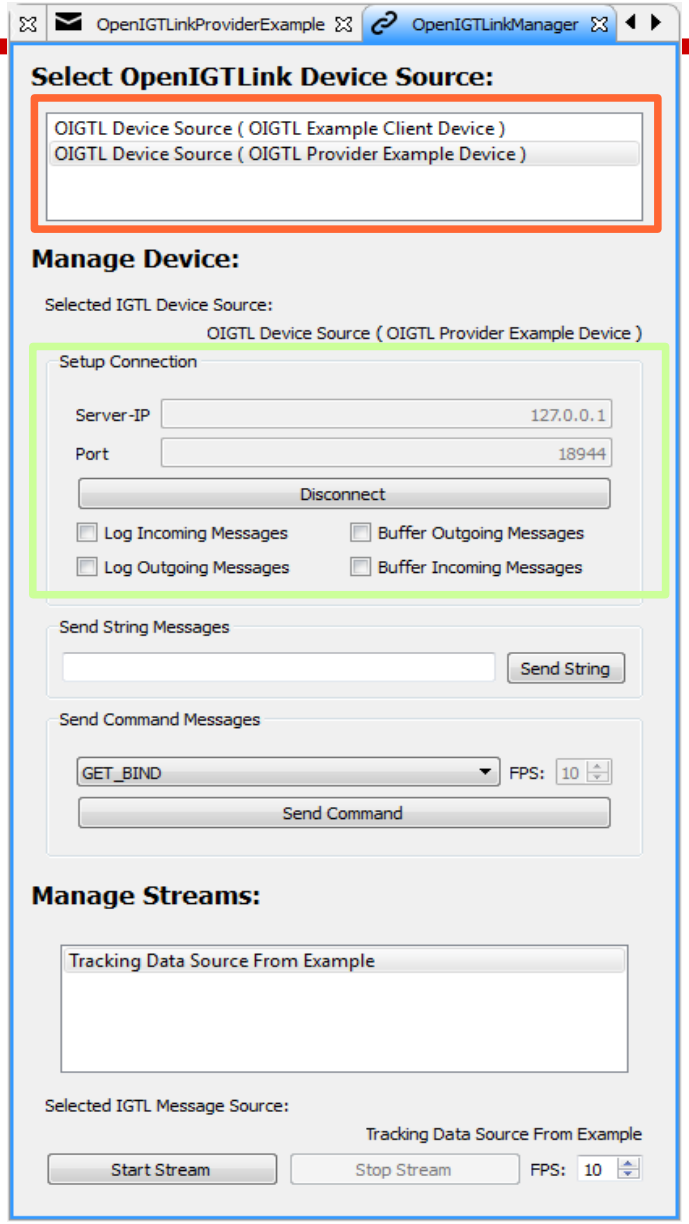
Implementation - Manager



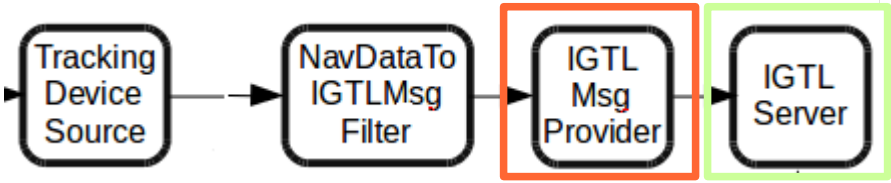
- List of all OIGTL device sources registered as µService



Implementation - Manager

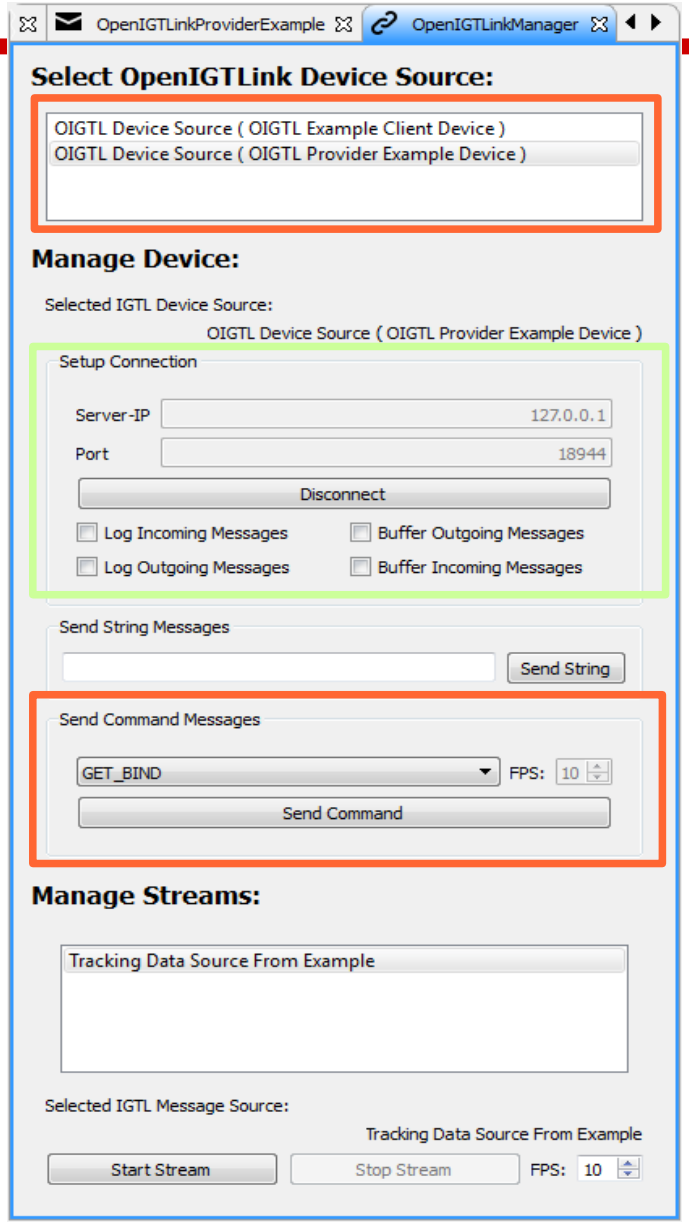


- List of all OIGTL device sources registered as µService

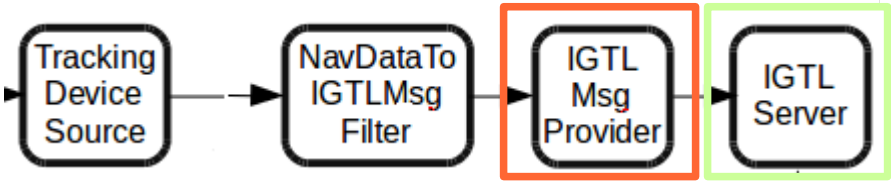


- Connect / Disconnect devices
- Turn On/Off logging
- Turn On/Off buffering

Implementation - Manager



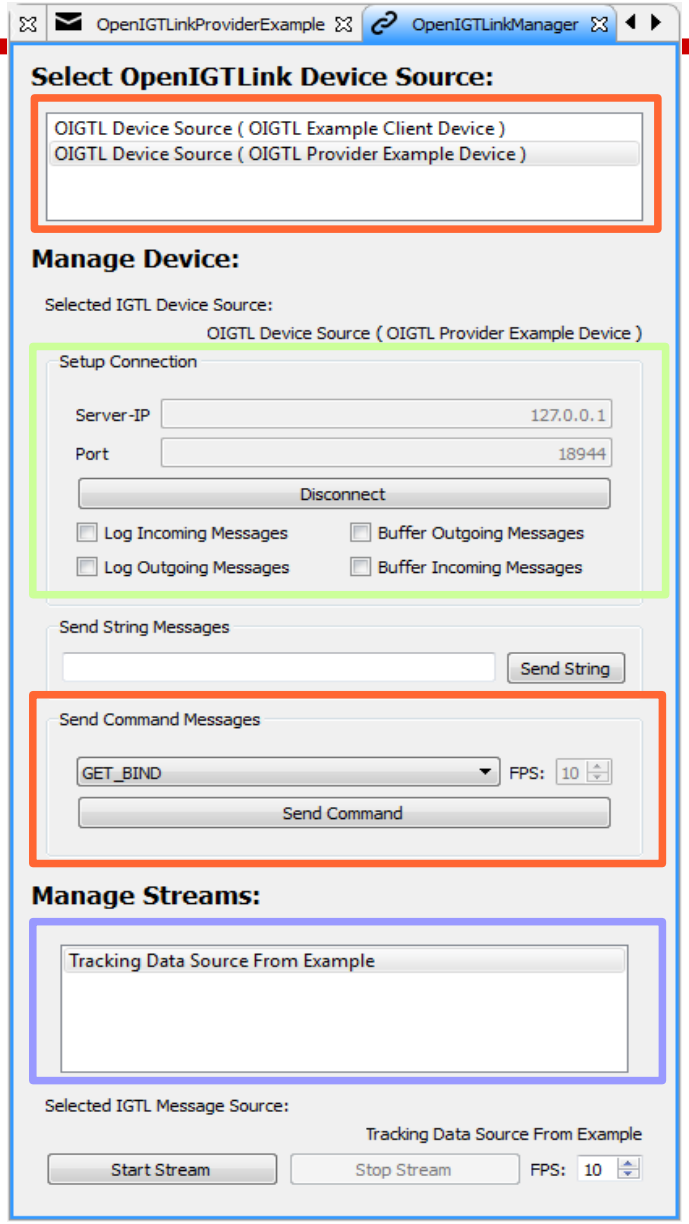
- List of all OIGTL device sources registered as µService



- Connect / Disconnect devices
- Turn On/Off logging
- Turn On/Off buffering

- Query single data (GET_*) or data streams (STT_*)
- Stop streams (STP_*)

Implementation - Manager



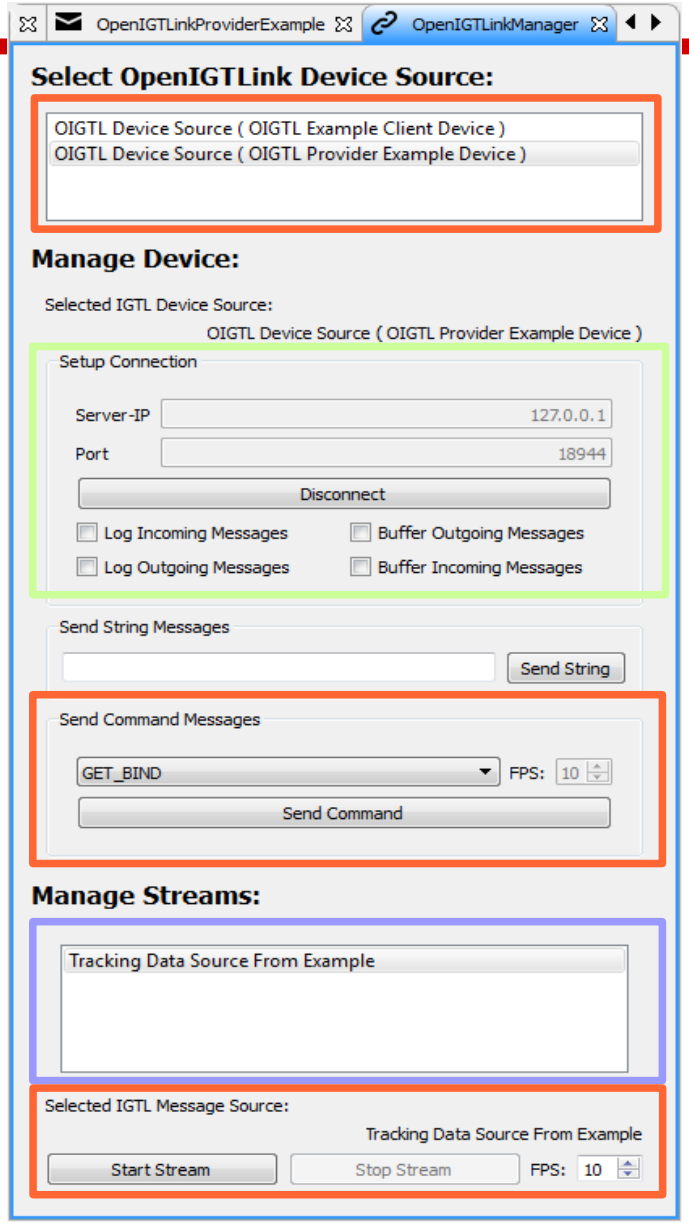
- List of all OIGTL device sources registered as µService

- Connect / Disconnect devices
- Turn On/Off logging
- Turn On/Off buffering

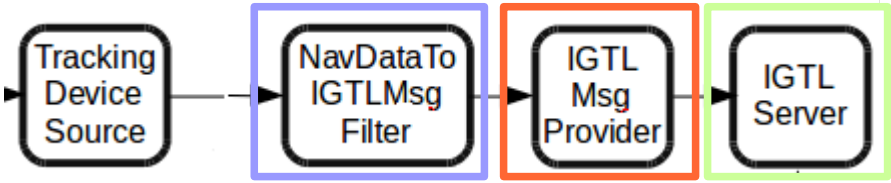
- Query single data (GET_*) or data streams (STT_*)
- Stop streams (STP_*)

- List of all OIGTL message sources registered as µService

Implementation - Manager



- List of all OIGTL device sources registered as µService



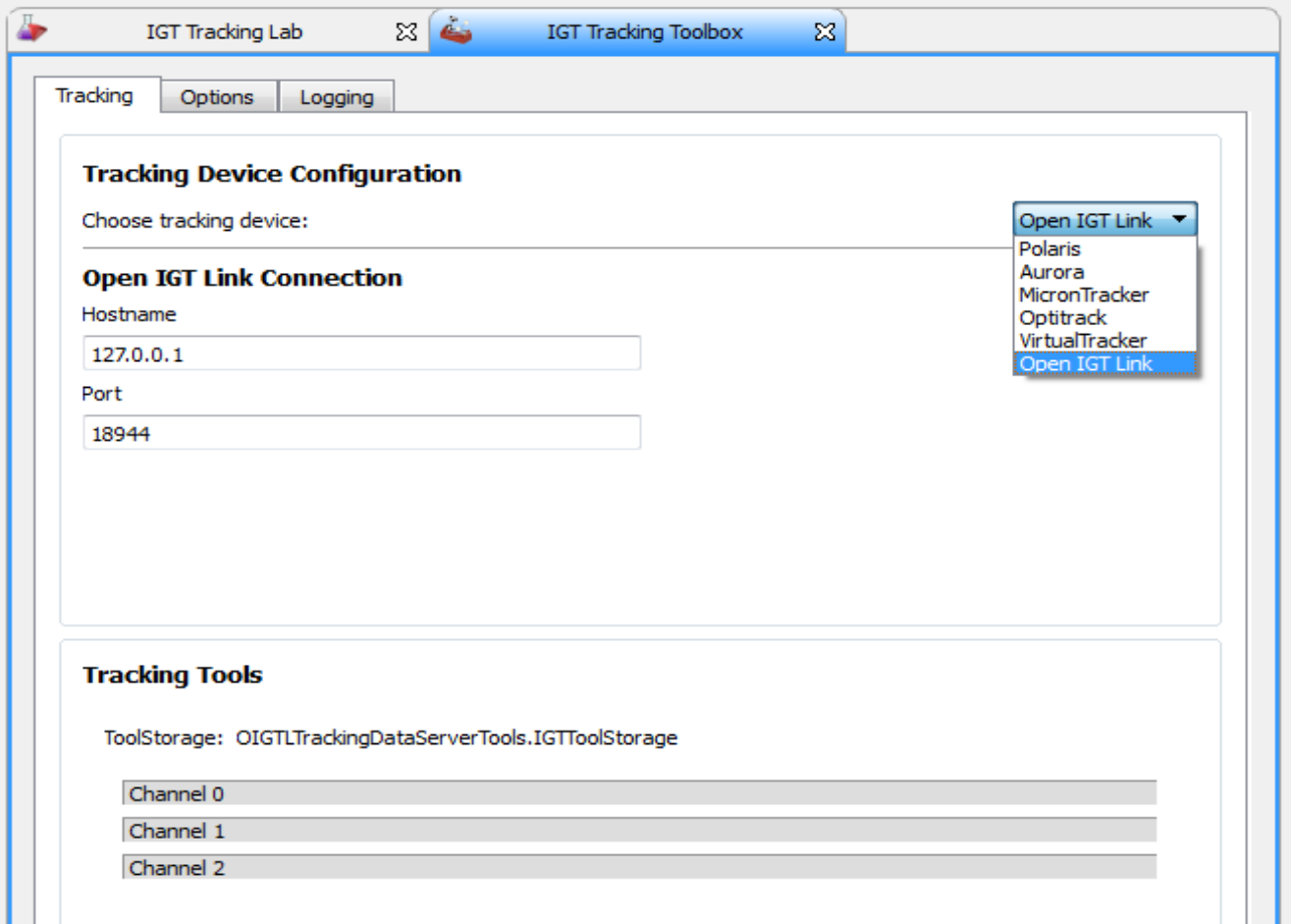
- Connect / Disconnect devices
- Turn On/Off logging
- Turn On/Off buffering
- Query single data (GET_*) or data streams (STT_*)
- Stop streams (STP_*)
- List of all OIGTL message sources registered as µService
- Streaming Control

Features

- Sending and receiving data to/from other OpenIGTLink devices
- Message buffering (configurable between queuing and non-queuing)
- Using custom data types
- Integration into the MITK processing pipeline
- Starting and stopping of OpenIGTLink message streams
- Implemented Query Concept

Work-in-Progress

- Integration into IGT- and US-GUI- Elements



Outlook

- Sending and receiving status/keep-alive-messages
- Performance tests
- Interoperability tests with other toolkits

Summary

- Foundation and basic functionality is implemented
- First examples and tests were performed
- OpenIGTLink is currently integrated into exisiting IGT and US components
- Interoperability and Performance tests will be performed

For further information check the API documentation of the MITK-OpenIGTLink module and the OpenIGTLink example plugins

Summary

- Foundation and basic functionality is implemented
- First examples and tests were performed
- OpenIGTLink is currently integrated into exisiting IGT and US components
- Interoperability and Performance tests will be performed

For further information check the API documentation of the MITK-OpenIGTLink module and the OpenIGTLink example plugins