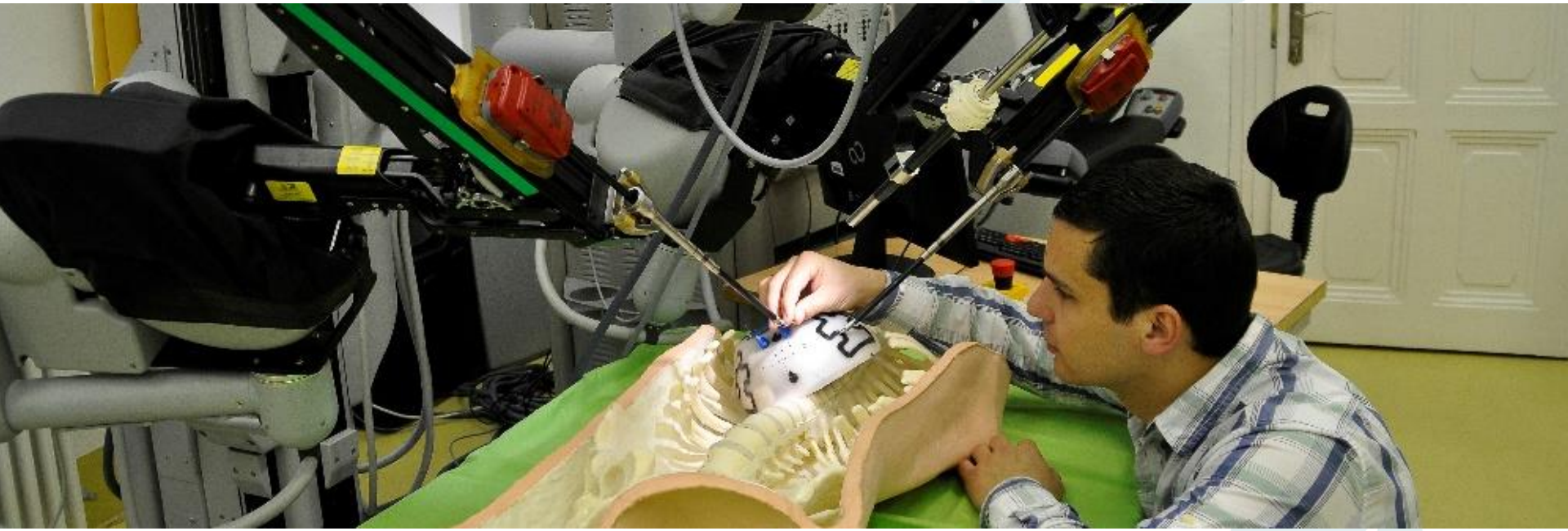
Safety Issues with Component-based Software Frameworks



Péter Galambos, Tamás Haidegger

ERF 2017

22nd March 2017, Edinburgh, UK

Component Frameworks

RTM – Robot Technology Middleware

- 2003 - , AIST – Japan (Noriaki Ando)
- Based on CORBA (OMG standard)
- OMG RTC 1.1 Standard

MRDS – MS Robotics Developer Studio

- 2006 - 2014, Microsoft, USA
- Based on MS DSSP (a lightweight SOAP)

ROS – Robot Operating System

- 2009 - ,Willow Garage, USA (today maintained by OSRF)
- TCPROS, UDPROS, XML-RPC

ROS 2 – Robot Operating System 2

- 2016 - , OSRF
- DDS (OMG Standard, but not Open Source!)

Maturity

- RTM: Japan only
- MRDS: Project stopped
- ROS:
  - Tens of thousands of users
  - In 2015 somewhere **between $100 and $200** million in venture capital has been raised by companies that use ROS.
  - **But, security, robustness and reliability are not considered in the design.**

Safety isseus

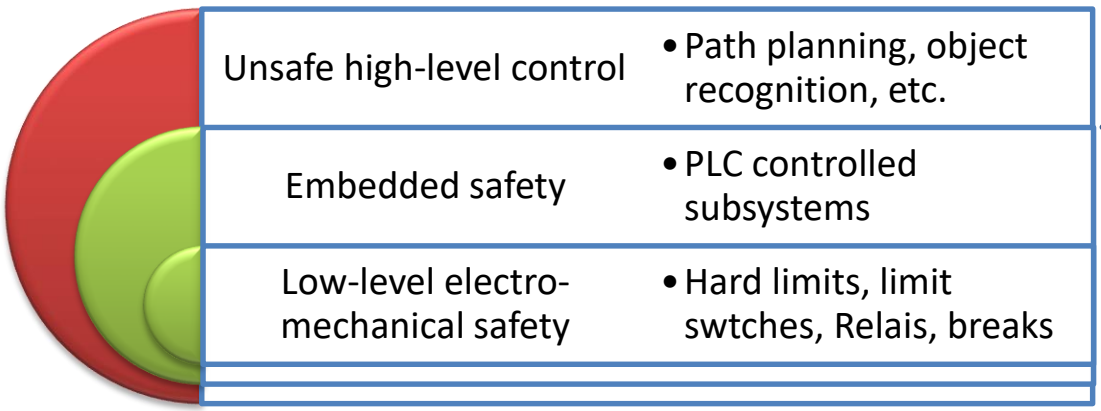
- IT Security (system integrity, vulnerability)
- Reliable communication (package loss)

Designs will abide by standards: “Our current software (based on ROS) doesn’t achieve safety, so when we’re interfacing with industrial robots, we always keep the controller, which contains the safeties.”

Shaun Edwards (SwRI)

ROS 2 and DDS

- DDS is widely used in missin critical distributed control application
- Industry standard **real-time interoperation middleware**
- Sophisticated QoS capabilities through transport configurations
  - Deadlines
  - Fault-tolerance
  - Synchronization
- Durability via inherent resilence mechanisms



A new leyer comes here:
„Safe" distributed control

Examples in medical robotics

Only research tools are developed so far

- **DaVinci Research Kit** (DVRK) -> ROS-based API over embedded RT controller
- **RAVEN II** -> Control software implemented over ROS + RT Linux (@ UW)
- **KIT Modular MIS environment** -> Fully ROS driven (A. Bihlmaier et al. @ KIT)
- **KUKA IIWA** stack (Salvatore Virga & Marco Esposito @ TUM)
- **Autonomous robotics surgery experiments (@ BARK)**
- **Surgical Robot Challenge** („Hamlyn challange”)
  - IIWA, DVRK, RAVEN
- And many more…

Conclusions

- Safety of current ROS-based systems cannot be guaranteed because of potentially unreliable communication and vulnerability
- Viable approaches
  - Safety-critical functions are always supervise by „reliable” controllers
  - Safety-critical functions are online monitored by a reliable watchdog
    - **Actual state of the nodes**
    - **Network delay and jitter**
    - **Frequency of messages**
  - **Improved structural safety through DDS**
- Development and operation workflow should handle
  - Software safety design
  - Validation
  - System-level verification
  - E.g., According to IEC 61508-3, IEC 62061

Message

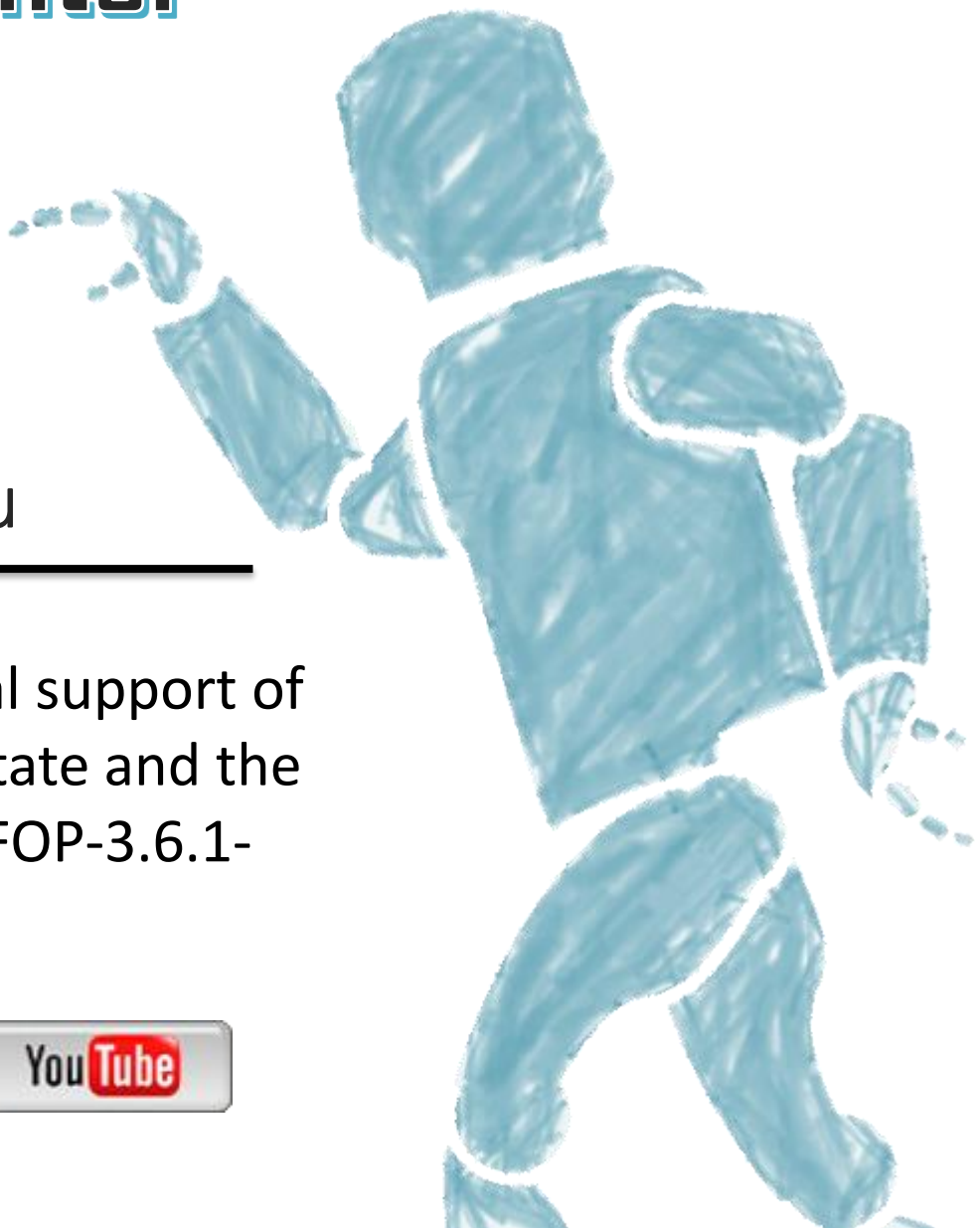
Safe distributed control should be considered in upcoming standards

Antal Bejczy Center for Intelligent Robotics

http://irob.uni-obuda.hu

We acknowledge the financial support of this work by the Hungarian State and the European Union under the EFOP-3.6.1-16-2016-00010 project.

Facebook Linkedin YouTube



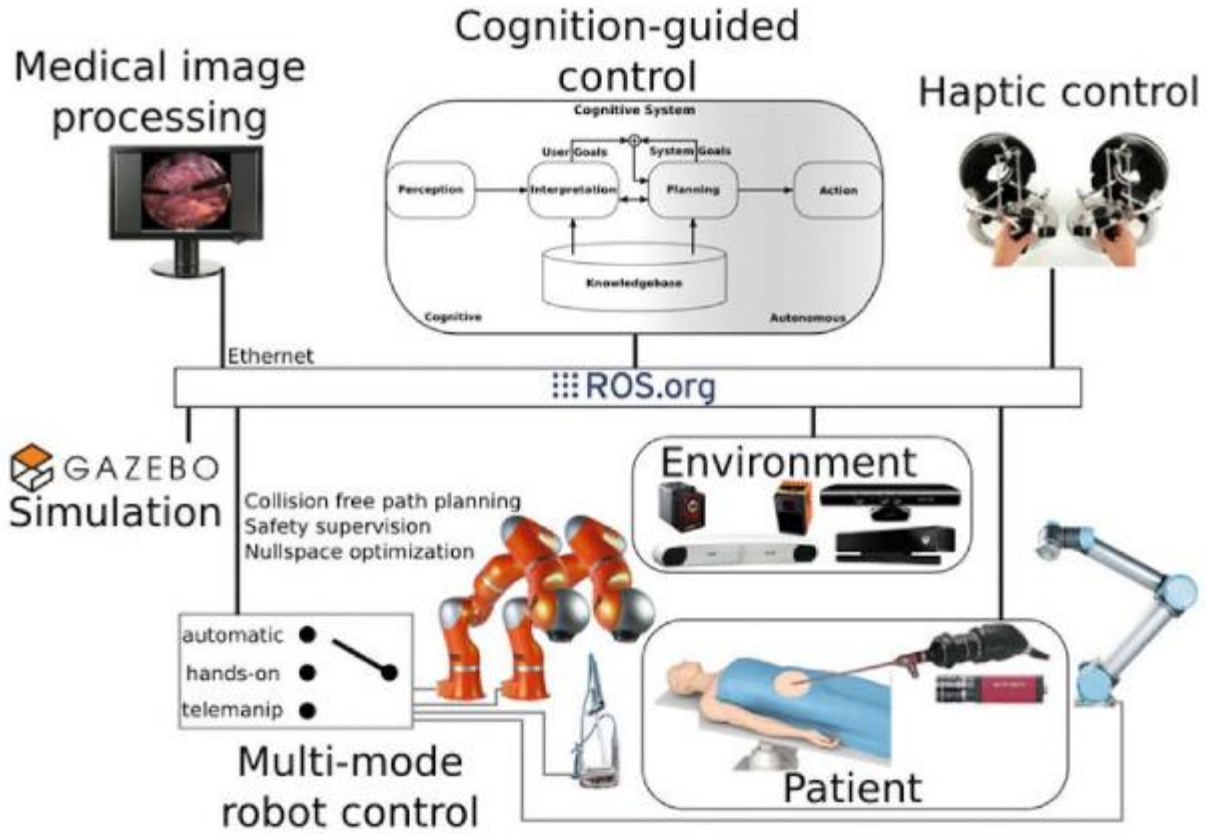
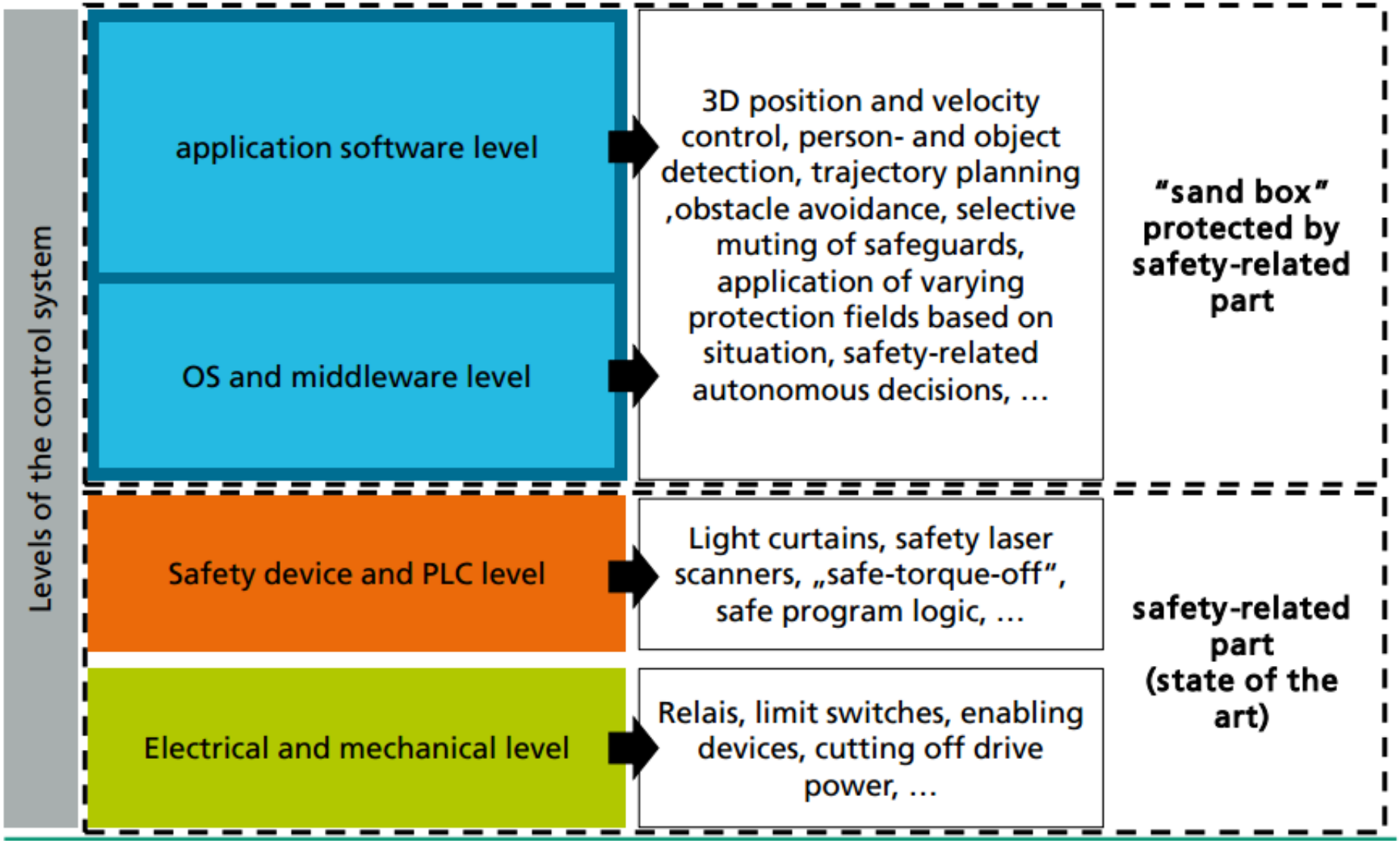


Fig. 1 Overview of our modular ROS-based research platform for robot-assisted minimally-invasive surgery (cf. [4]). The system contains several components to perceive the environment, the patient and the user. Physically actions can be executed by different robots. Interchangeability of components is essential, i.e. higher level algorithms should not have to know which robot, tracking system or camera is used. The whole setup also exists as a virtual model for the robotics simulator Gazebo. Thus algorithms can be evaluated in simulation on a single host without the necessity of accessing the real lab setup. This benefits both researchers and students by reducing the problem of scheduling lab access

A. Bihlmaier et al. (KIT)



Theo Jacobs, Fraunhover IPA