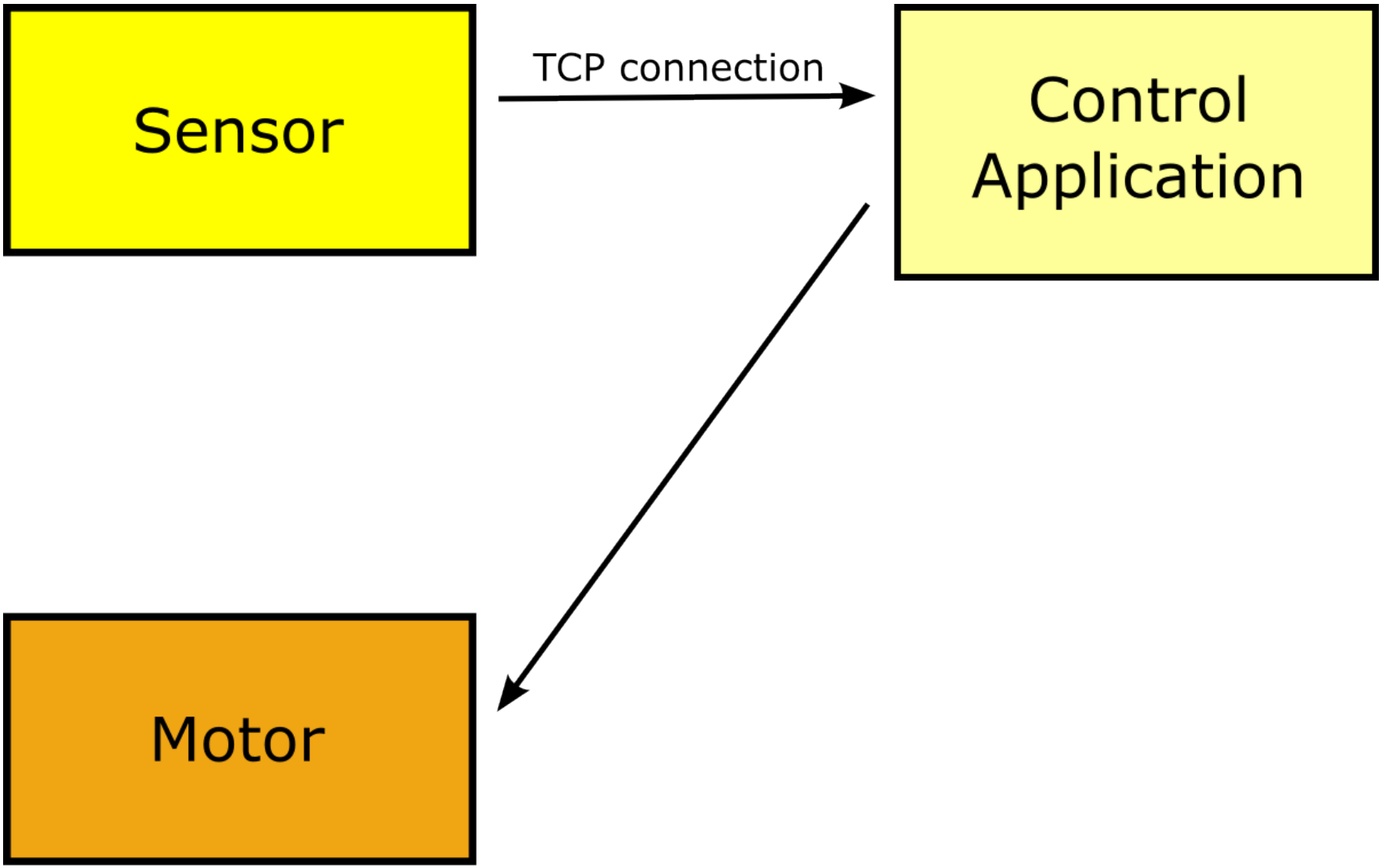
M.Vajnar, M. Sojka, P. Píša

Czech Technical University in Prague

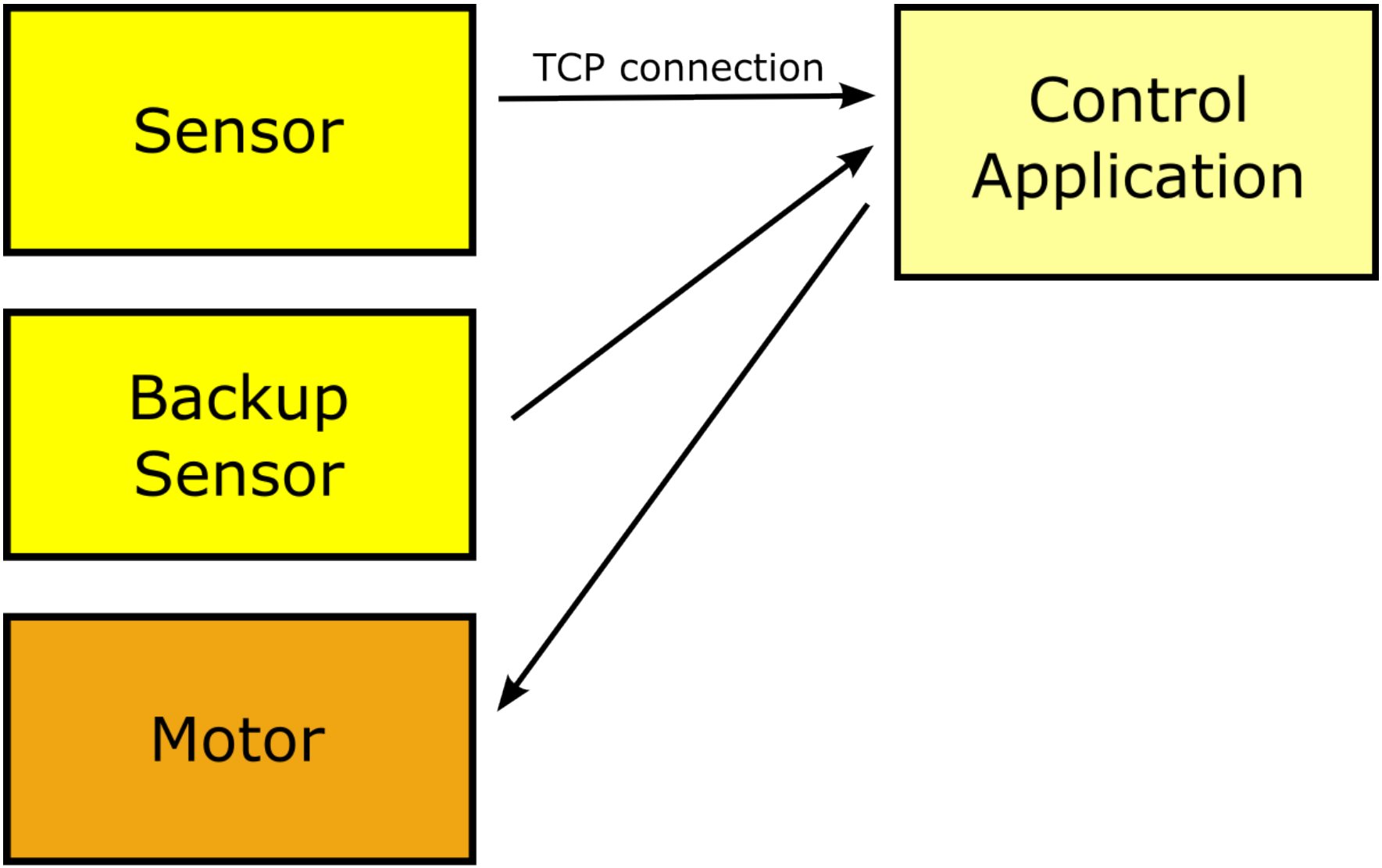
Porting of Real-Time Publish-Subscribe Middleware to Android

RTLWS15, Lugano-Manno

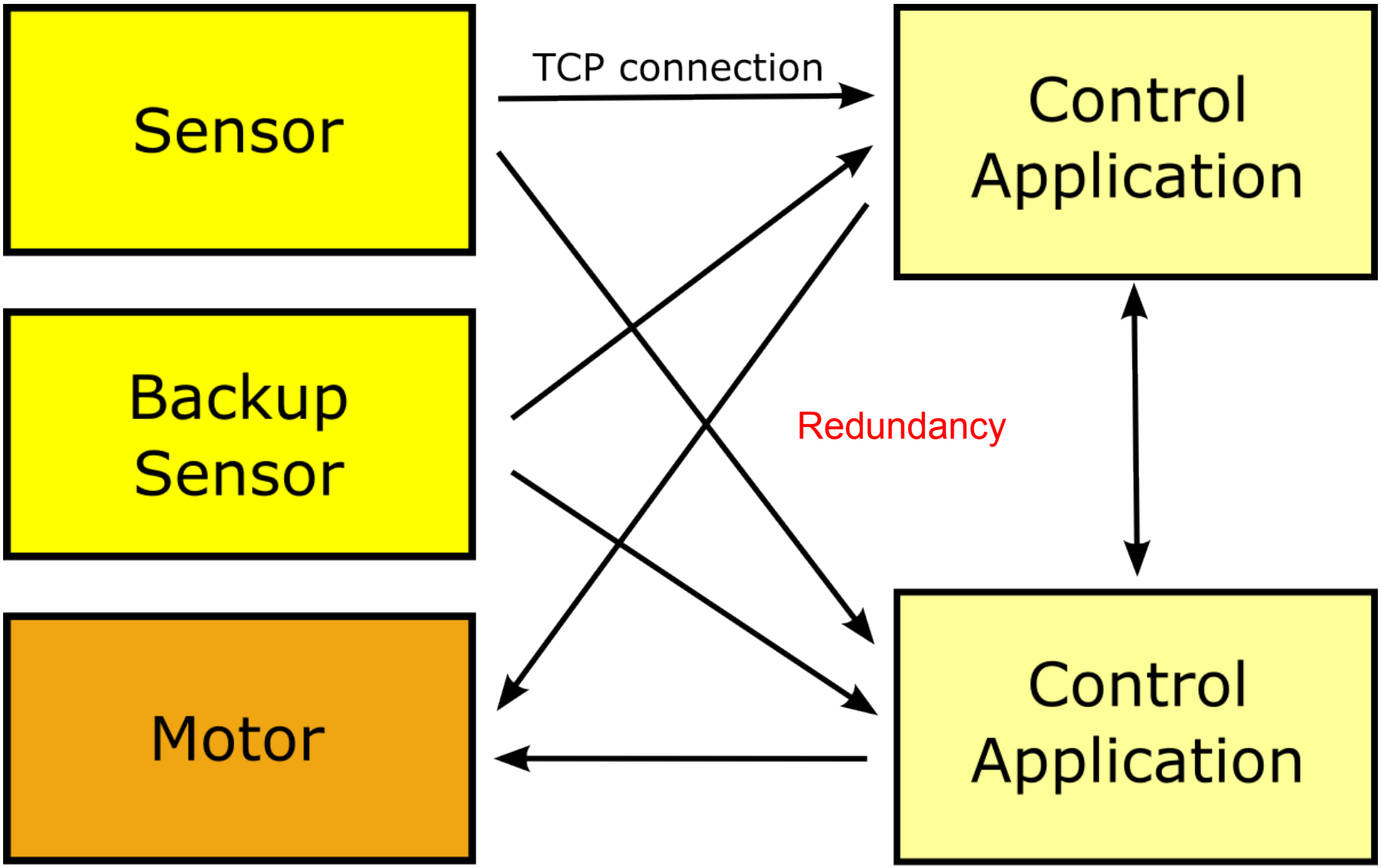
Distributed applications – problems



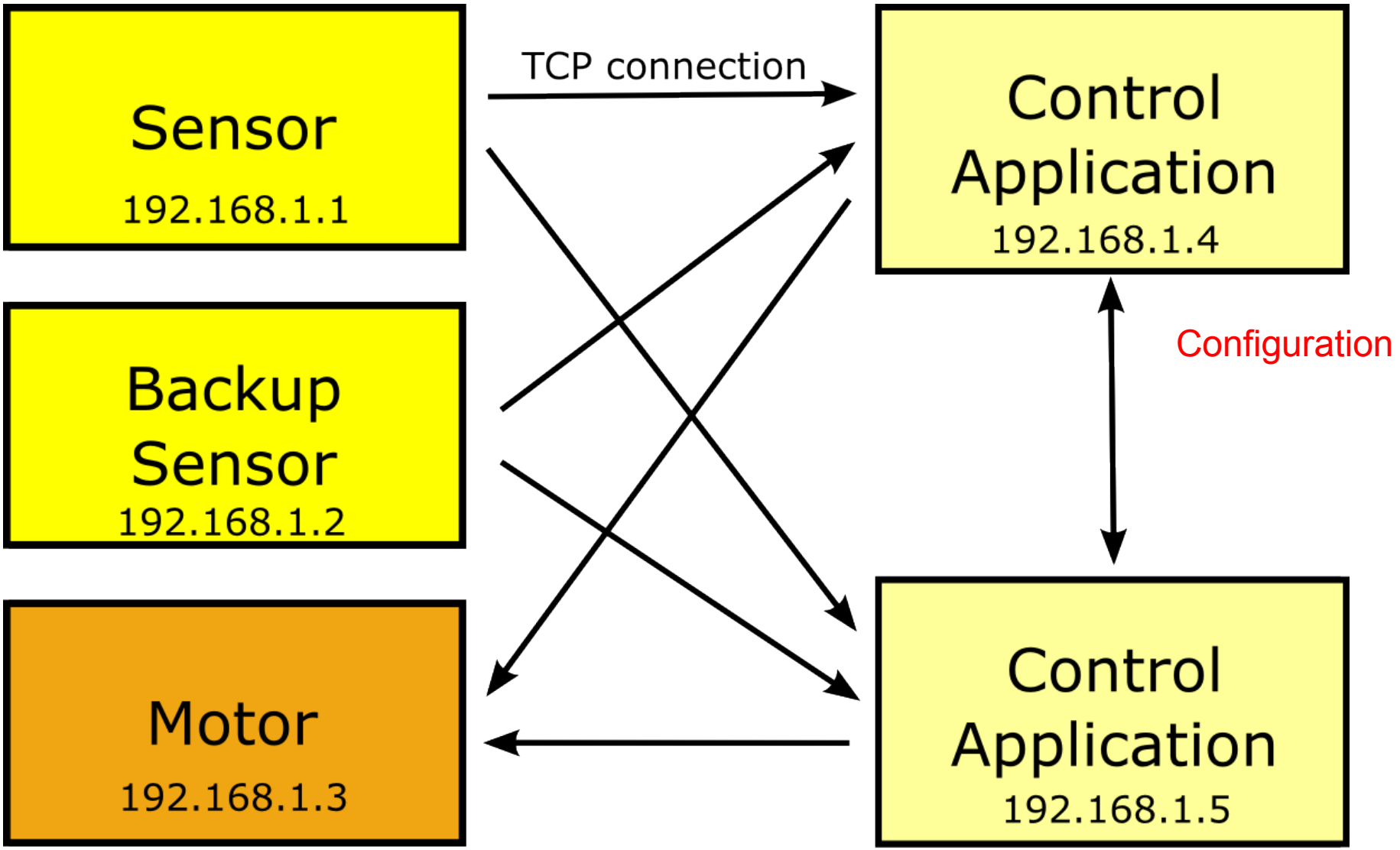
Distributed applications – problems



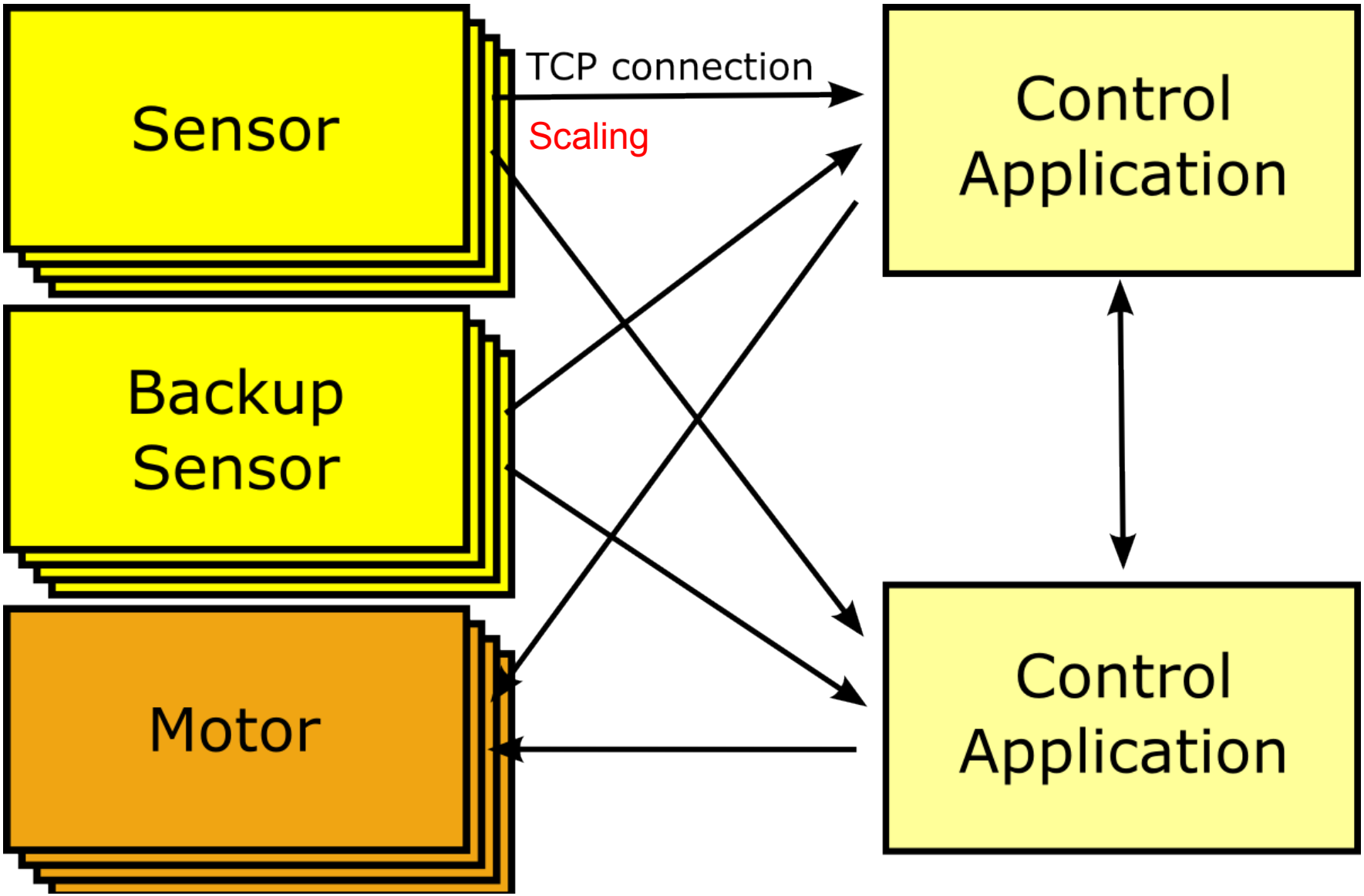
Distributed applications – problems



Distributed applications – problems



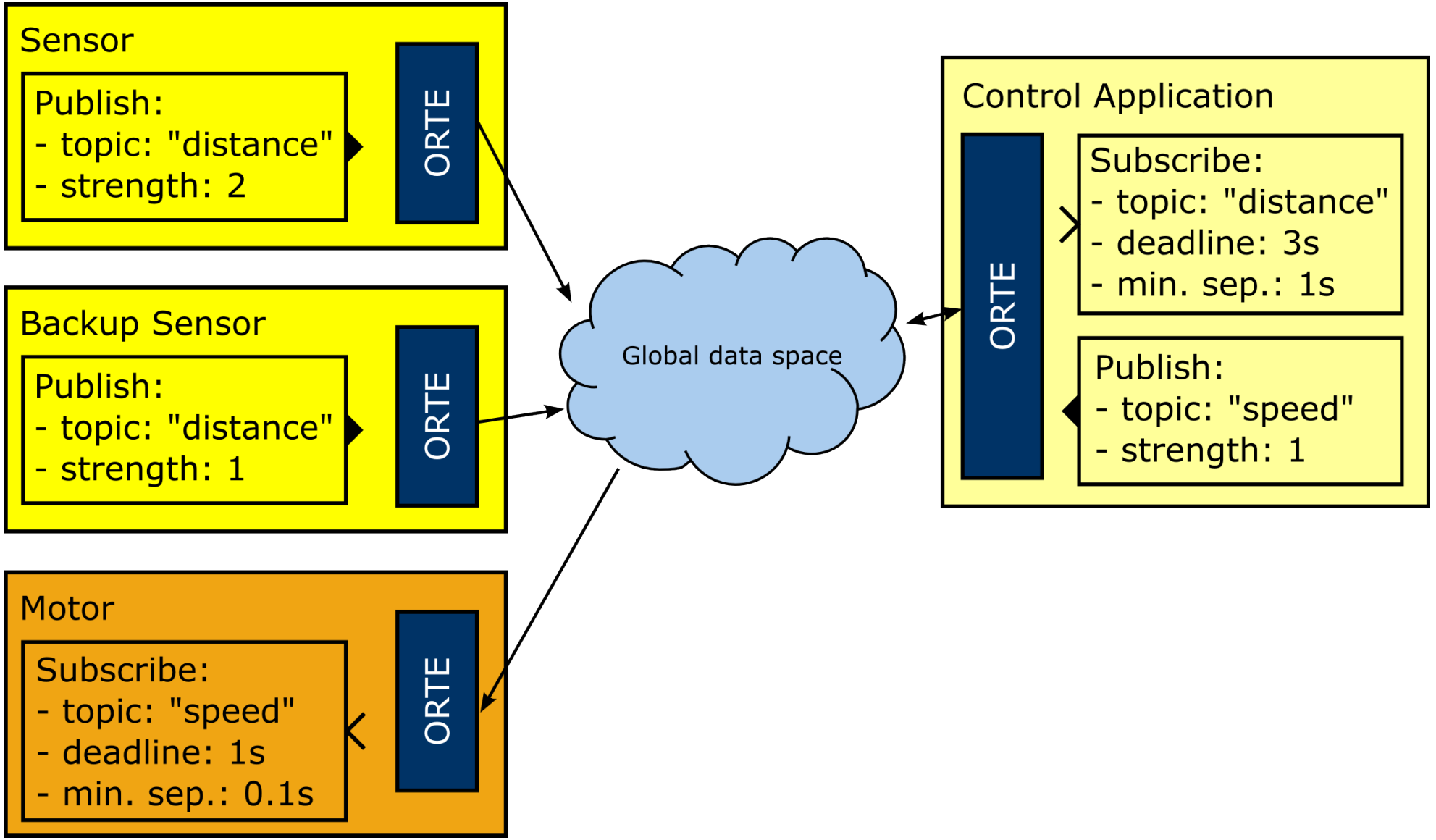
Distributed applications – problems



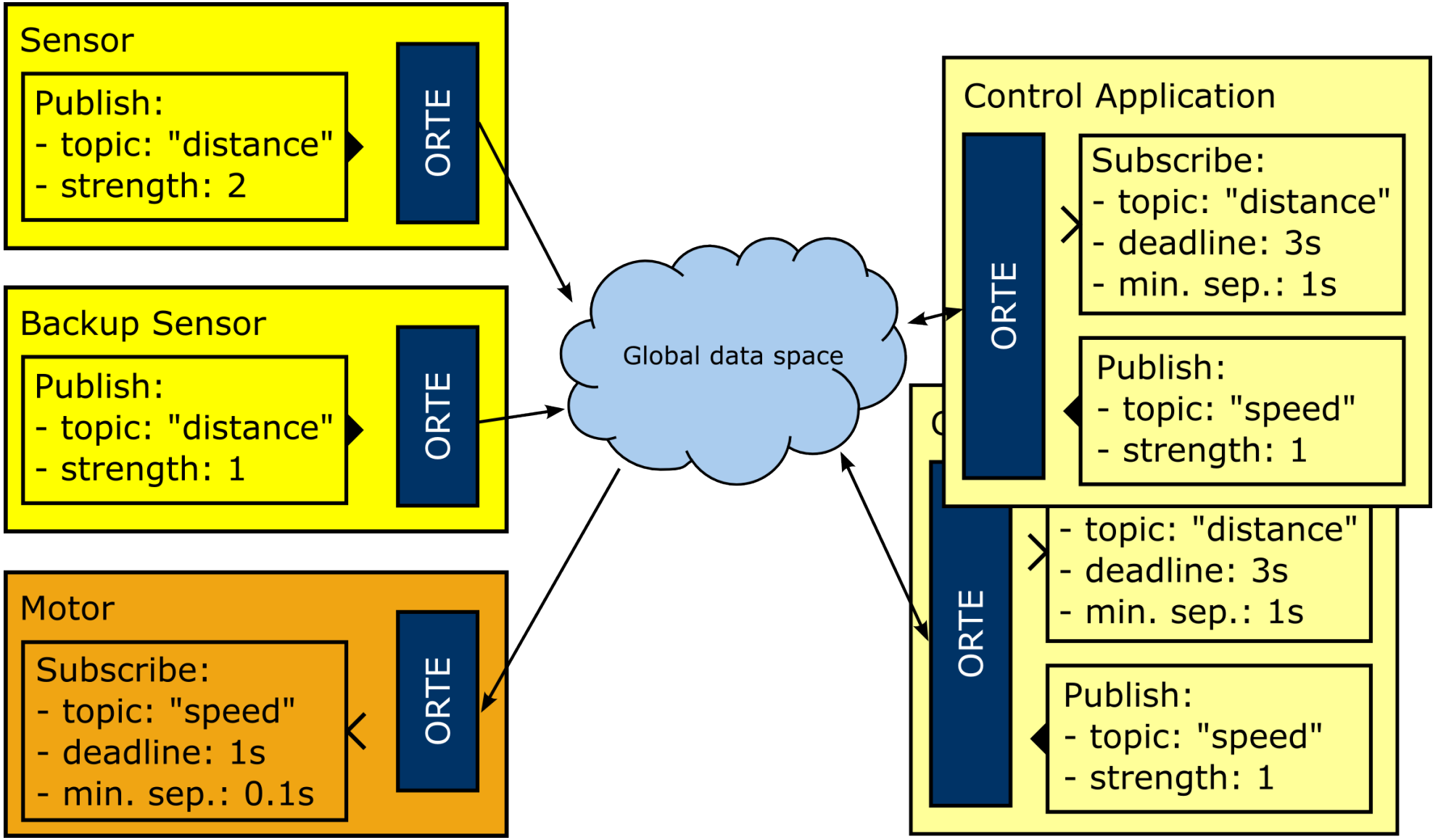
Outline

- Data-Centric Publish Subscribe model
- Real-Time Publish-Subscribe protocol
- ORTE middleware
- Android porting
- Demo

Data-Centric Publish-Subscribe application model



Data-Centric Publish-Subscribe application model



How to implement this? RTPS.

- Real-Time Publish-Subscribe protocol
- OMG standard – interoperability wire protocol
- Transport independent part
- Transport specific part (UDP => control of timing)
- Implementation freedom
  - Simple implementation, high network bandwidth demand
  - Complex implementation, optimized network bandwidth.
- Utilizes multicast communication (one to many communication)
- Offers best-effort and reliable communication
- CDR encoding (endianing)
- Parts
  - Data exchange protocol
  - Discovery protocol

ORTE – Open Real-Time Ethernet

- Our open-source implementation of the RTPS protocol
- One of the first few implementations that supported standardization
- Uses custom API instead of the DDS API (another OMG standard)
- C language, based on POSIX
- IDL compiler to generate (de)serialization functions
- Suported platforms
  - Linux, FreeBSD, Mac OS X, Solaris
  - Windows (MinGW, Cygwin, ReactOS)
  - RTEMS
  - BlackBerry
  - Android

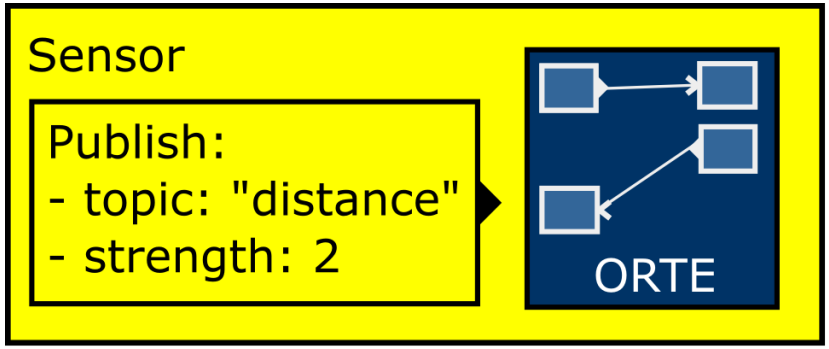
Minimal ORTE application

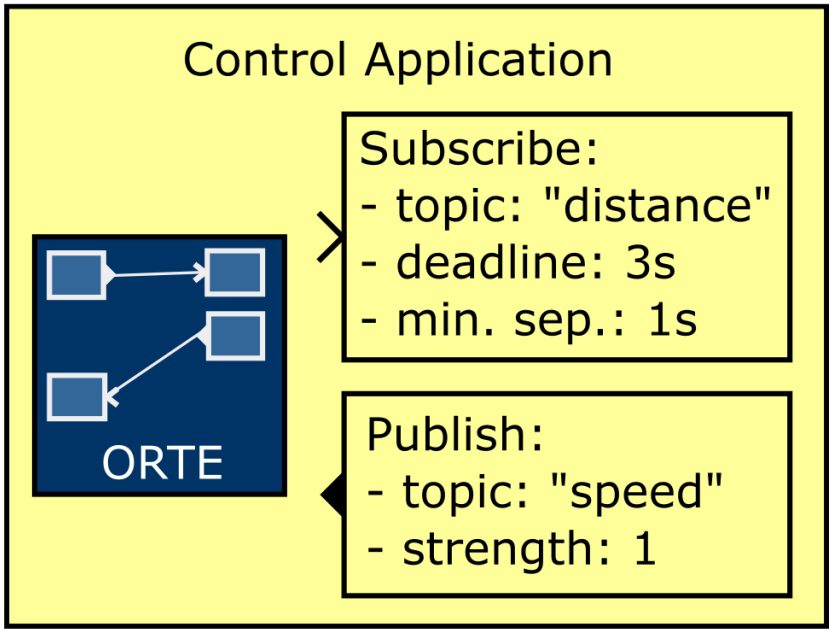
```
ORTEInit();
d=ORTEDomainAppCreate(0,NULL,NULL,0);
ORTETypeRegisterAdd(d,"HelloMsg",NULL,
                    NULL,NULL,64);
p=ORTEPublicationCreate(
     d,
     "Example HelloMsg",  // Topic
     "HelloMsg",          // Type
     &instance2Send,      // Data buffer
     TIME(3,0),           // Persistance
     1,                   // Strength
     NULL,                // Callback
     NULL,                // Parameters
     NULL);
...
instance2Send = 123;
ORTEPublicationSend(p);
```

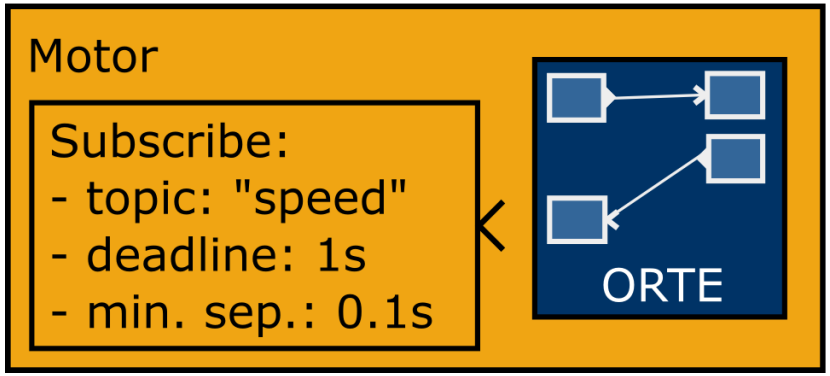
```
ORTEInit();
d=ORTEDomainAppCreate(0,NULL,NULL,0);
ORTETypeRegisterAdd(d,"HelloMsg",NULL,
                    NULL,NULL,64);
s=ORTESubscriptionCreate(
    d,
    IMMEDIATE,
    BEST_EFFORTS,
    "Example HelloMsg",     // Topic/
    "HelloMsg",             // Type
    &instance2Recv,         // Data buffer
  &deadline,
    &minimumSeparation,
    recvCB,                 // callback
    NULL,
    IPADDRESS_INVALID);     // Multicast

void recvCB(const ORTERecvInfo *info,
            void *instance, void *param) {
  switch (info->status) {
    case NEW_DATA:
      printf("%s\n",instance); break;
    case DEADLINE:
      printf("deadline occurred");   break;
  }
}
```

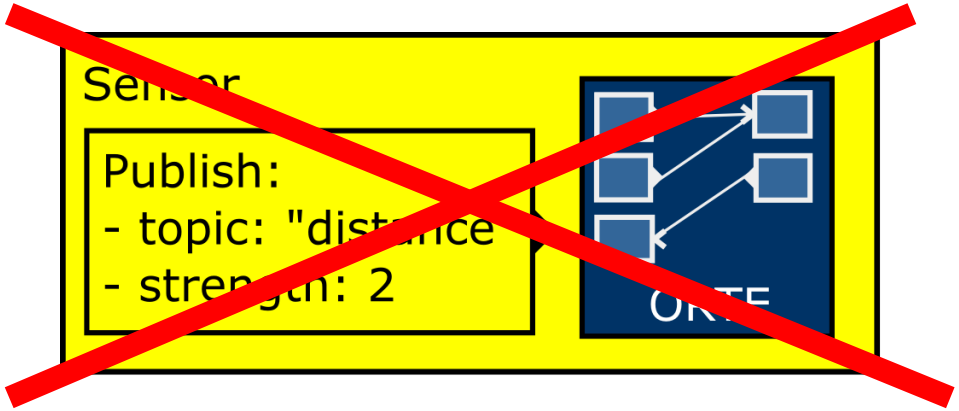
RTPS/ORTE internals

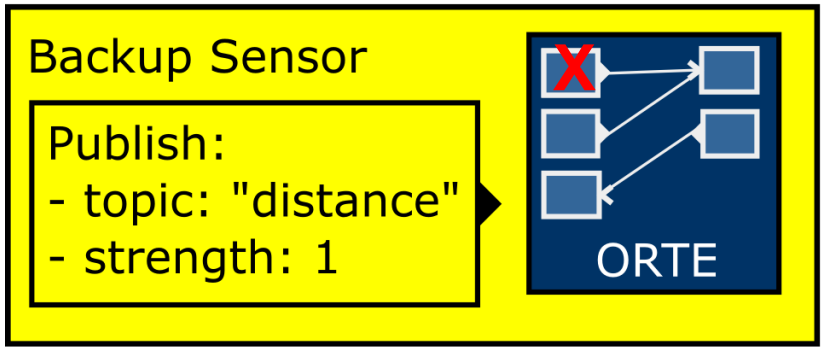


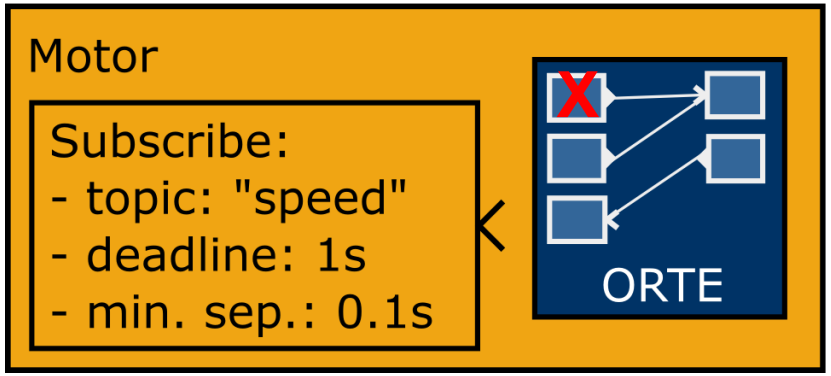


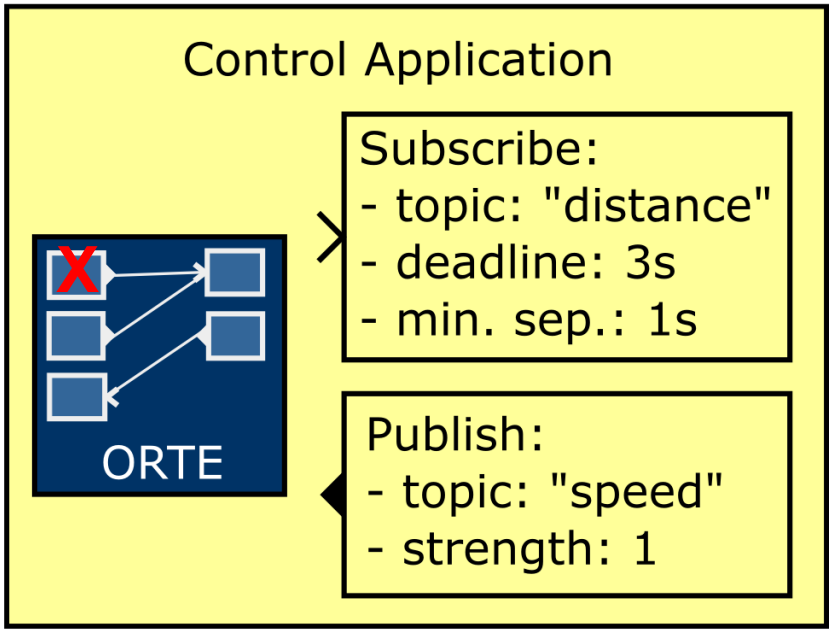


RTPS/ORTE internals



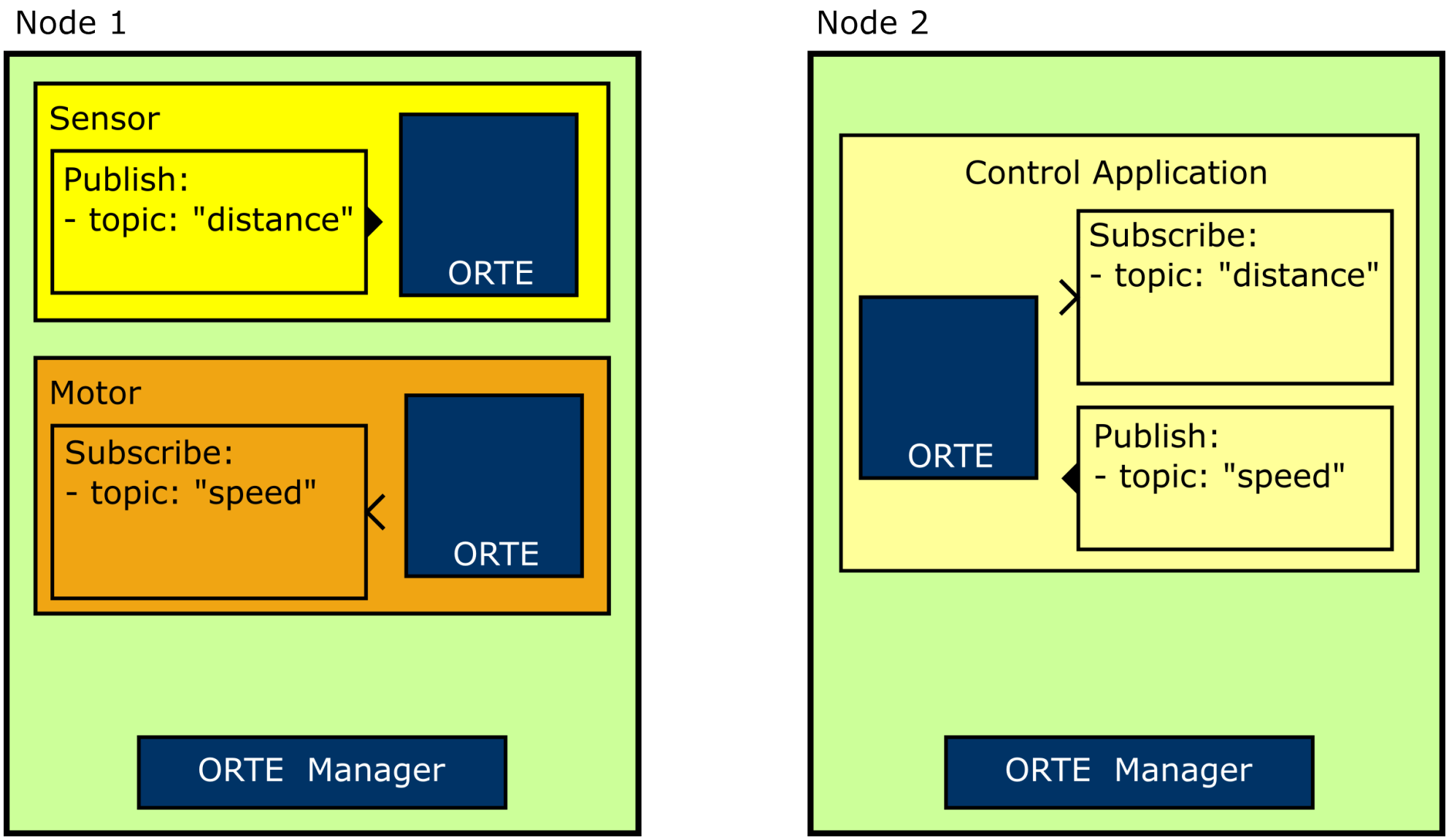




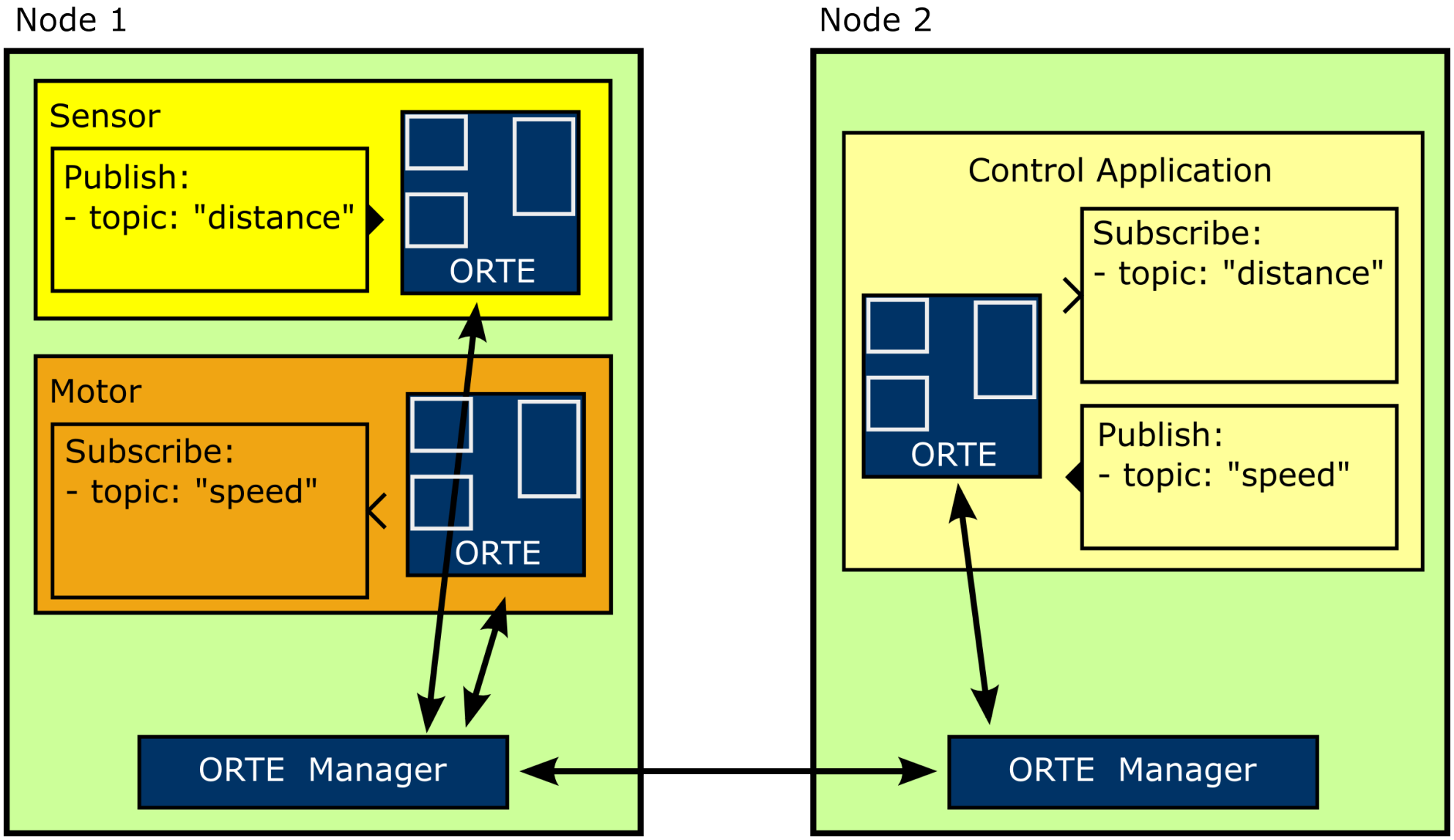


- Dynamic addition/removal of applications
- No single point of failure
- Fault tolerant applications

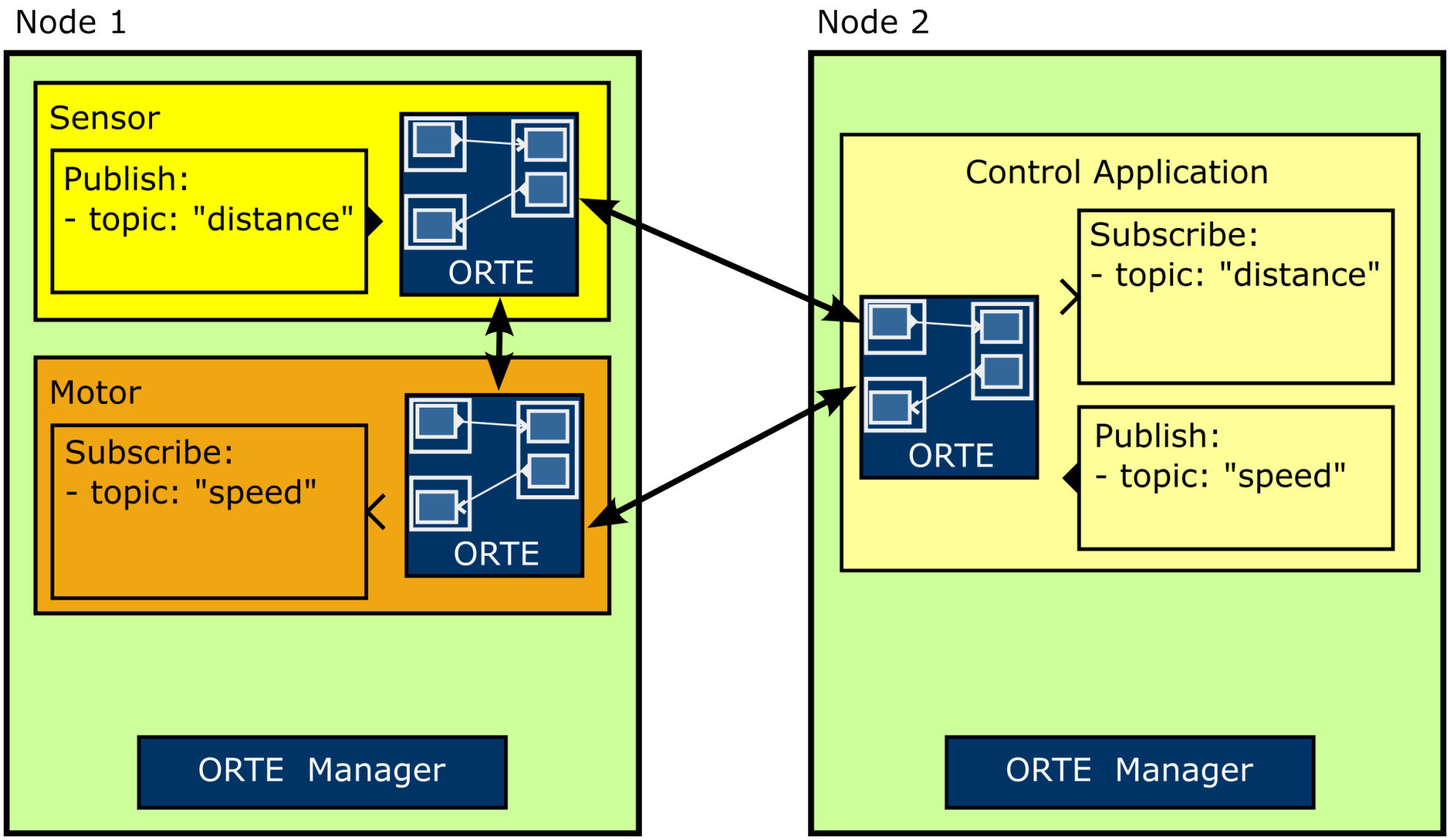
ORTE Discovery Protocol



Participant discovery



Endpoint discovery



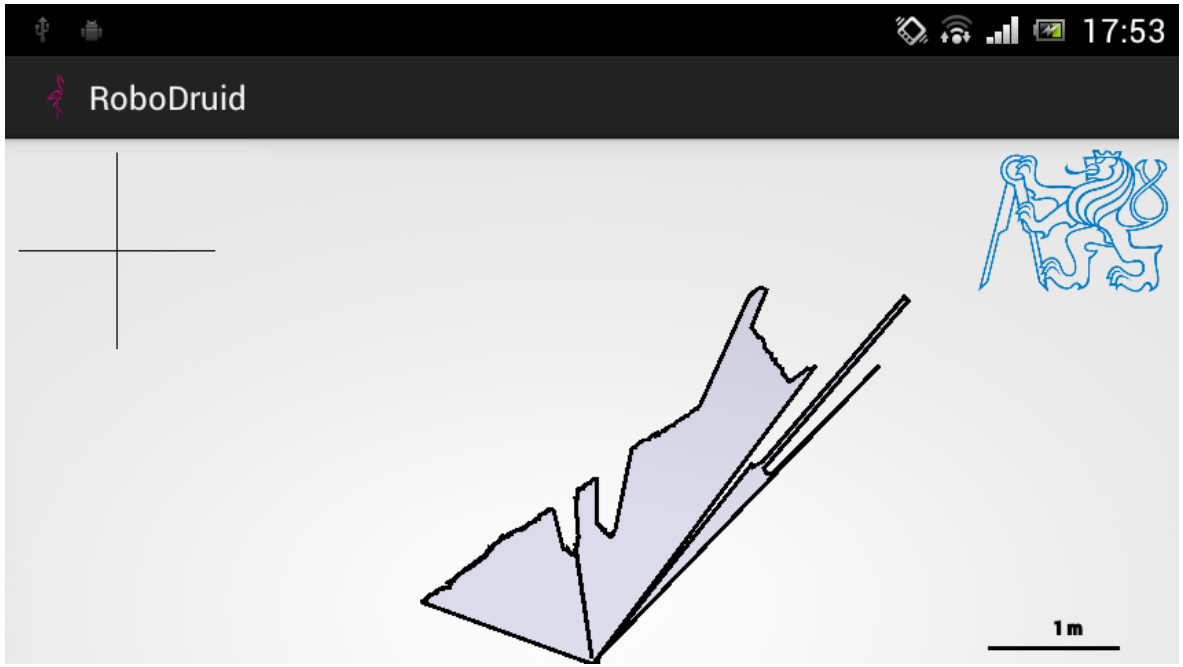
Android

- Android is a Linux-based operating system developed by Google
- It runs on devices ranging from mobile phones and tablets to home media centers and digital cameras

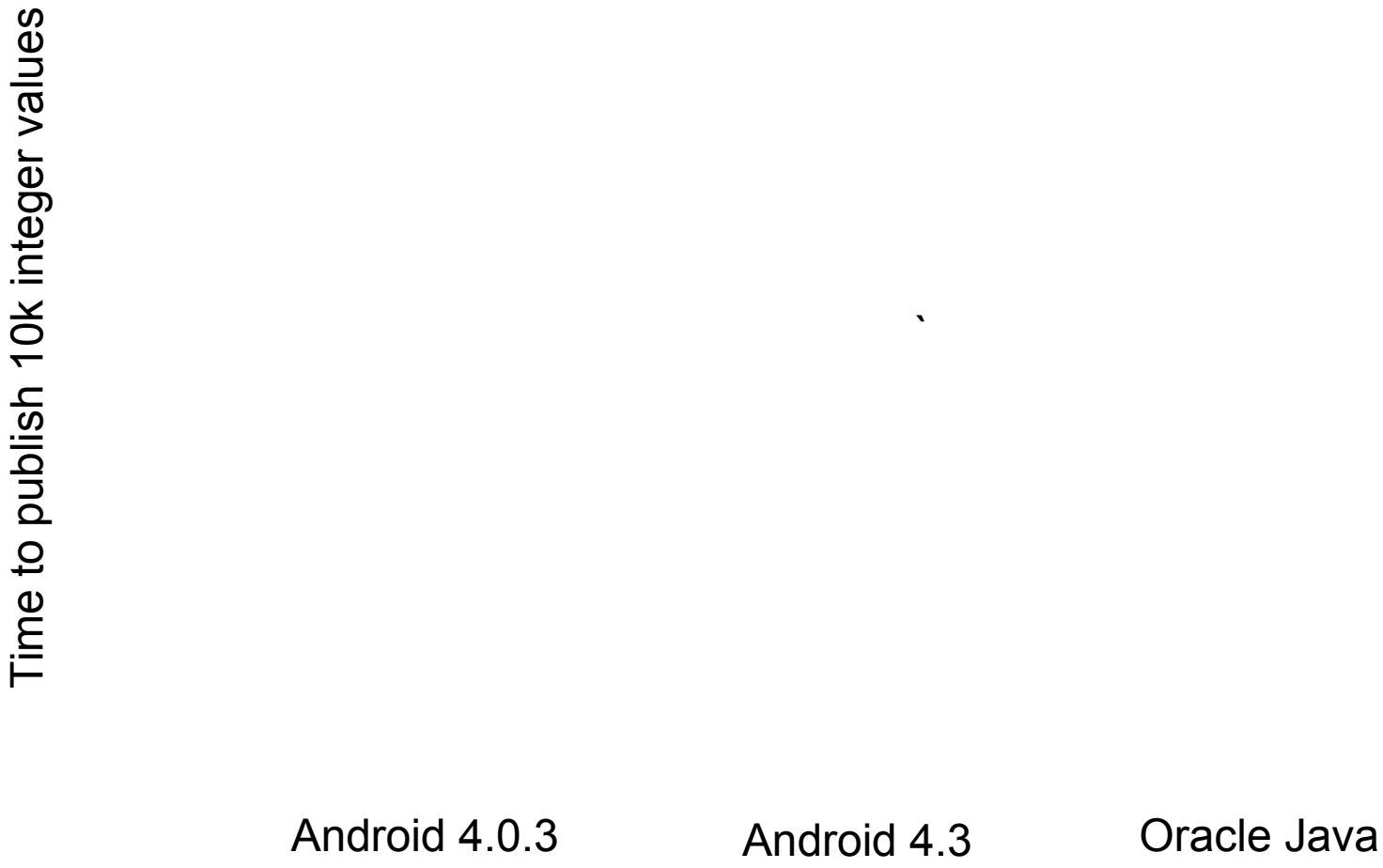
- Applications developed in Java or C/C++ using Google toolkits
  - Android Software Development Kit – Java
  - Android Native Development Kit – C/C++

Porting ORTE to Android

- **Overview:**
  - Update Java Native Interface (JNI) wrapper and make it Android compatible
  - Fix bugs that have not demonstrated themselves under the Oracle's VM
  - Add support for Android build system
  - Make Java version of ORTE Manager application to overcome problems with execution and termination of native processes



Performance comparison



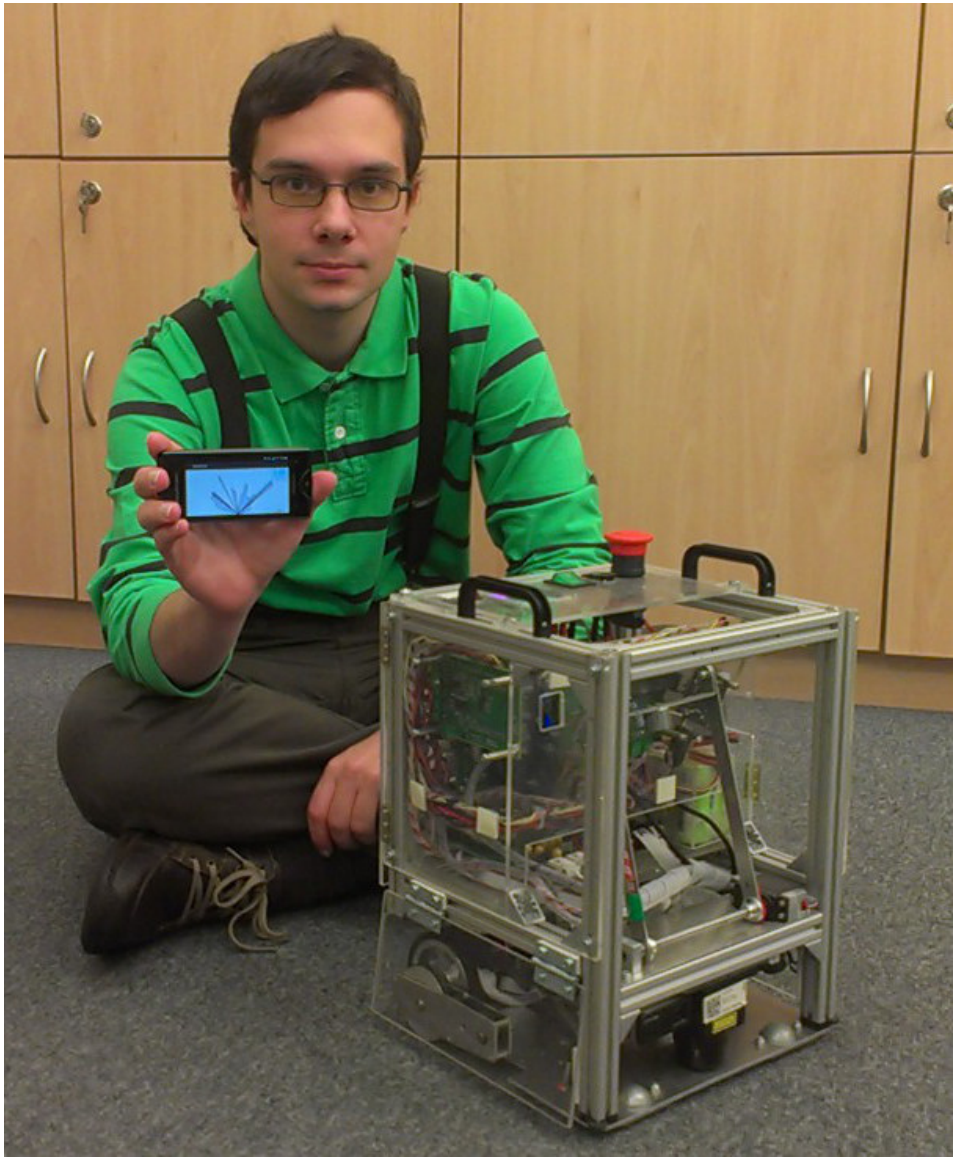
- Reliable publisher and subscriber
- Publish as fast as possible
- Baseline: C publisher, C subscriber

Future development

- **Make the ORTE compliant with the latest RTPS standard**
  - **Data with key**
    - New type of data objects, that allows to distribute a set of data instances under a single topic
    - The key is used to distinguish between instances
  - **Discovery protocol**
    - Manager is replaced with the Simple Participant Discovery Protocol and the Simple Endpoint Discovery Protocol integrated into an user application
  - **Data fragmentation**
    - Allow a fragmentation of big data instances and sent them as multiple messages.
- **Security**

Demo

- Application for remote control of a robot
- Monitors robot's state (battery voltage, speed vector, output of Laser Range Finder)
- Controls robot's motion (direction and speed)



Conclusion

- **ORTE can simplify both development and deployment of distributed applications**
- **We have successfully ported the ORTE library to Android**
- **http://orte.sourceforge.net/**

Thank you!