Technische Universität Berlin

Use of TUB test facilities TWIST

CREW Training days 2nd edition. Advanced course

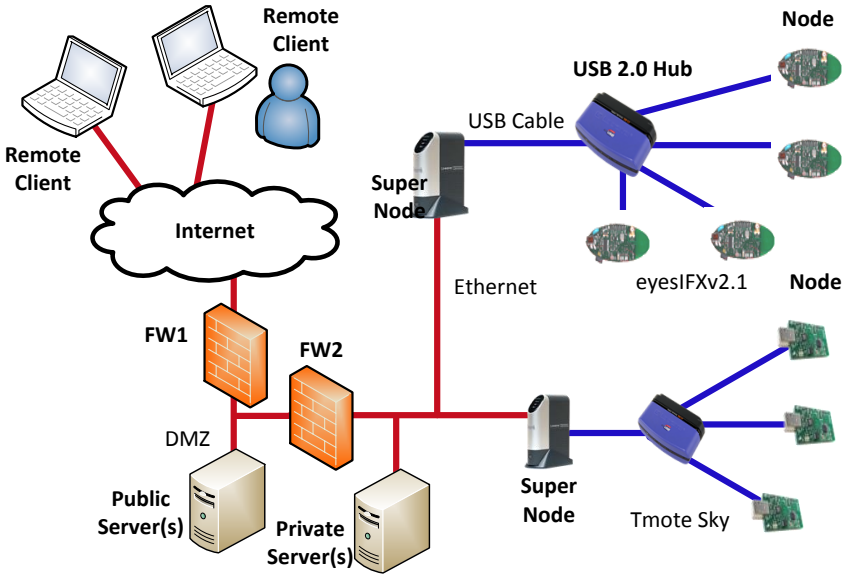
Mikolaj Chwalisz
(chwalisz@tkn.tu-berlin.de)

January 15, 2014

TKN Telecommunication Networks Group

Outline

TWIST Components [1]



Motivation

TWIST

TKN Wireless Indoor Sensor network Testbed

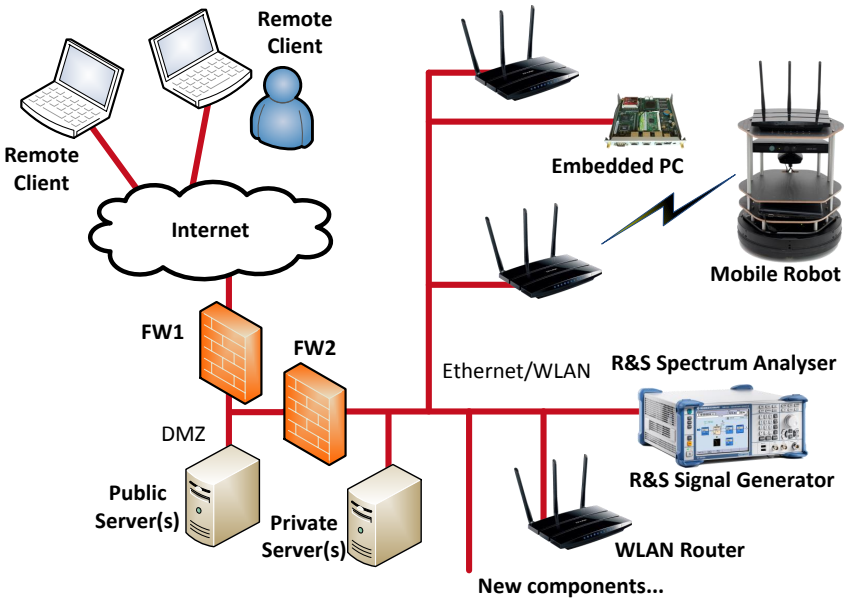
New requirements

- Interference generation
- Experiment with heterogeneous technologies
- Flexible experimentation with different hardware
- Simple way of adding new elements

CREW Testbed Components

1. WLAN Routers
   - TP-Link TL-WDR4300
2. Embedded PC's
   - Alix Boards
   - Broadcom Wireless LAN card
   - Wireless MAC Processor [2]
3. Mobile robot
   - Turtlebot II robotic platform
   - Kobuki mobile base
   - Laptop,
   - Microsoft Kinect 3D camera
   - Robot Operating System (ROS)

CREW Testbed Components



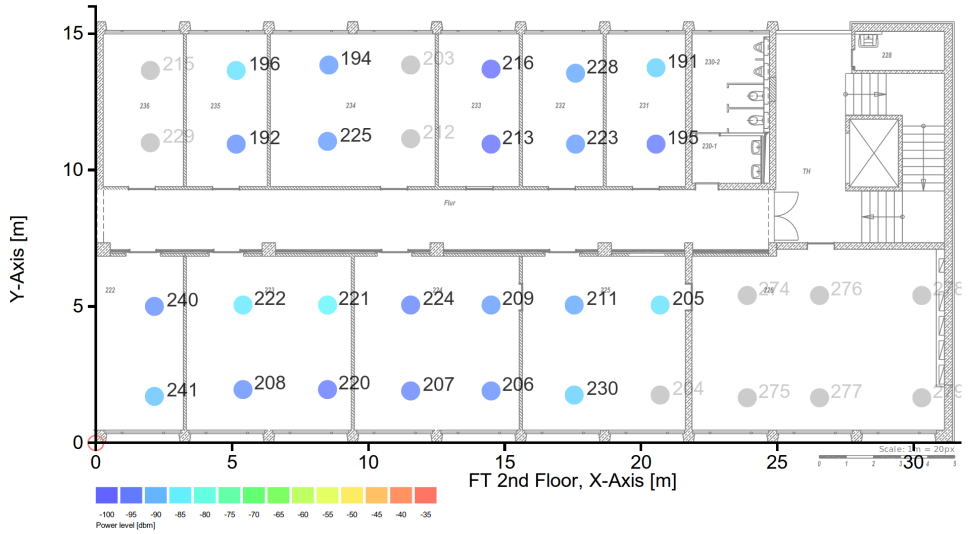
Demo Description

- Wireless Sensor Nodes for live spectrum monitoring
- Use embedded PC's in home environment scenario
- Run the mobile robot

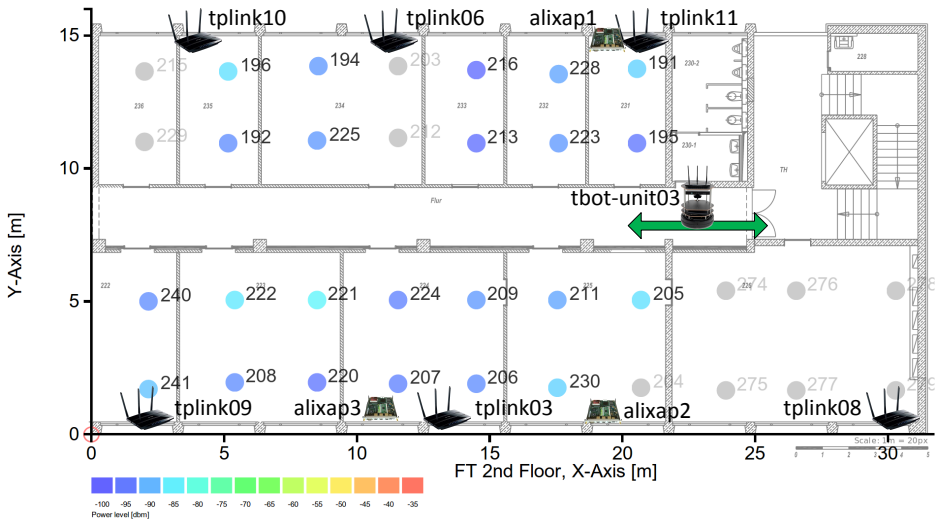
Connection

- Open VPN connection (`openvpn ovpnclient/CREWdemo.NetworkManager.openvpn`)
- Direct connection to all nodes in the network
- Connect to crewserver (`ssh guest@192.168.10.200` password: `crewguest`)

Spectrum Map



Node locations

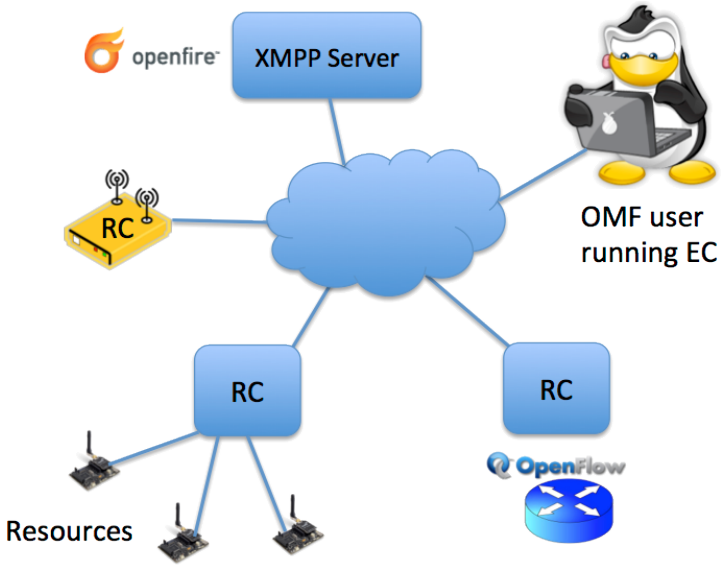


▸ DNS name resolution

WLAN Routers

- 5GHz band infrastructure for mobile robot
- 2.4GHz for experiments
- On-going work to enable OMF 6.0 based control
- Instalment on 2nd floor done, 3rd and 4th needs to be done

OMF 6.0 [3]



WLAN Interference generation

```
133 onEvent(:ALL_UP_AND_INSTALLED) do |event|
134   info "Environment has started..."
135   group("Server").startApplications   # Start ENV server
136   group("WiSpyNode").startApplications
137   # group("FsvNode").startApplications
138   after 15.seconds do
139     every 15.seconds do
140       info "Email Client started for 1s..."
141       group("Email_Client").startApplications
142       after 1.seconds do
143         group("Email_Client").stopApplications
144       end
145     end
146   end
147   after 20.seconds do
148     every 100.seconds do
149       info "Video Client started for 75s..."
150       group("Video_Client").startApplications
151       after 75.seconds do
152         group("Video_Client").stopApplications
153       end
154     end
155   end
156   after 5.seconds do
157     every 60.seconds do
158       info "Data Client started for 22s..."
159       group("Data_Client").startApplications
160       after 22.seconds do
161         group("Data_Client").stopApplications
162       end
163     end
164   end
165   after 320.seconds do
166     info "Environment has stopped..."
167     Experiment.done
168     allGroups.stopApplications
169     info "Collect wispy data by using:"
170     info "scp crewserver:/nfs/alixap/opt/data/#{OmfEc.experiment.id}* ."
171     info "cp /tmp/#{OmfEc.experiment.id}* ."
172   end
173 end
```

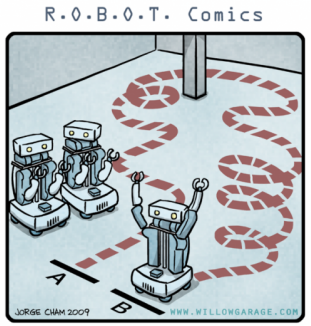
WLAN Interference generation

1. Install `omf_ec` on the machine
2. Use the one on `crewserver`
3. Start the script `omf_ec exec crewTrainingDays/iperfED_ENV.rb`
4. Finish experiment (wait or cancel manually)

Motivation

New capabilities

- Ability to place hardware wherever one wants over the building
- Particularly interesting for expensive equipment
- Ability to have a defined motion within the experiment that can be reproduced e.g. handover scenarios



"HIS PATH-PLANNING MAY BE SUB-OPTIMAL, BUT IT'S GOT FLAIR."

Limitations

- No interest in becoming real robotics guys
- No interest in deploying external localization hardware

TWISTbot

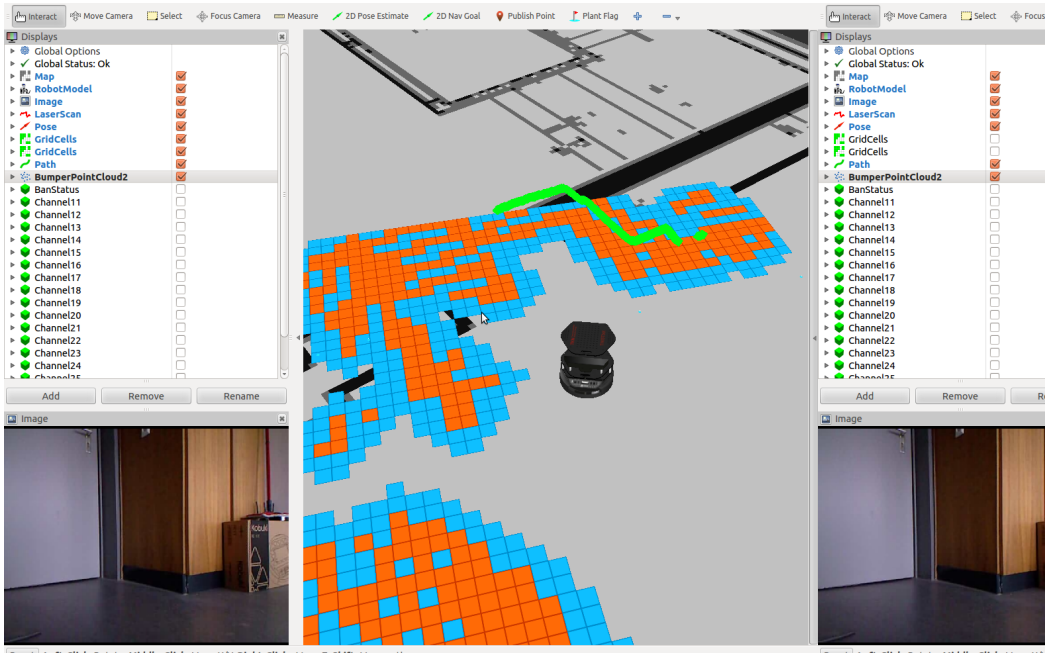
Turtlebot II robotic platform [4]

- Kobuki - mobile base
- Asus Netbook
- TP-Link WDR-4300 router
- Microsoft Kinect 3D camera

Robot Operating System (ROS) [5]

- Open source
- Large and active community
- Distributed system
- Implements the publish-subscribe model

ROS Native control: rviz



Other experimentation components

- R&S FSV7 Spectrum Analyser
- R&S SMBV100A Signal Generator
- Ethernet socket in every room in the building
- Any new Ethernet enabled devices

TWIST Demo Script

1. Prerequisites
   - Log in into TWIST: `https://www.twist.tu-berlin.de:8000/`
   - User: `crew_training_days` Password: `crtrda321`
2. Install custom sensor node image
   - Jammer application on one
   - Spectrum sensing application on all remaining nodes
3. Run visualization
4. Interact with a sensor network node in realtime
   - Get the spectrum sensing information
   - Control the jammer

Bibliography I

[1] V. Handziski, A. Köpke, A. Willig, and A. Wolisz, "Twist: a scalable and reconfigurable testbed for wireless indoor experiments with sensor network", in *Proc. of the 2nd Intl. Workshop on Multi-hop Ad Hoc Networks: from Theory to Reality, (RealMAN 2006)*, Florence, Italy, 2006.

[2] I. Tinnirello, G. Bianchi, P. Gallo, D. Garlisi, F. Giuliano, and F. Gringoli, "Wireless mac processors: programming mac protocols on commodity hardware", in *INFOCOM, 2012 Proceedings IEEE*, 2012, pp. 1269–1277. DOI: 10.1109/INFCOM.2012.6195488.

[3] T. Rakotoarivelo, M. Ott, I. Seskar, and G. Jourjon, "OMF: a Control and Management Framework for Networking Testbeds", in *SOSP Workshop on Real Overlays and Distributed Systems (ROADS '09)*, Big Sky, USA, 2009, p. 6.

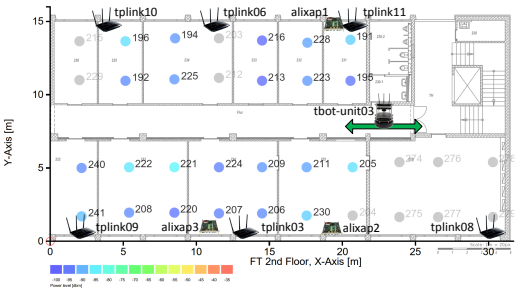
[4] *Turtlebot II Documentation*, 2013. [Online]. Available: `http://wiki.ros.org/Robots/TurtleBot`.

[5] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source Robot Operating System", in *ICRA workshop*, 2009.

DNS names

DNS name	IP address
crewserver	192.168.10.200
alixap1	192.168.10.209
alixap2	192.168.10.210
alixap3	192.168.10.211
tplink03	192.168.10.214
tplink06	192.168.10.217
tplink08	192.168.10.205
tplink09	192.168.10.206
tplink10	192.168.10.207
tplink11	192.168.10.208
tbot-unit03	192.168.10.12

Back to node locations



2.4 GHz ISM channels