iMinds w-iLab.t

Hands-on: running OMF experiments on w-iLab.t Zwijnaarde

Vincent Sercu
Pieter Becue
Stefan Bouckaert
Bart Jooris

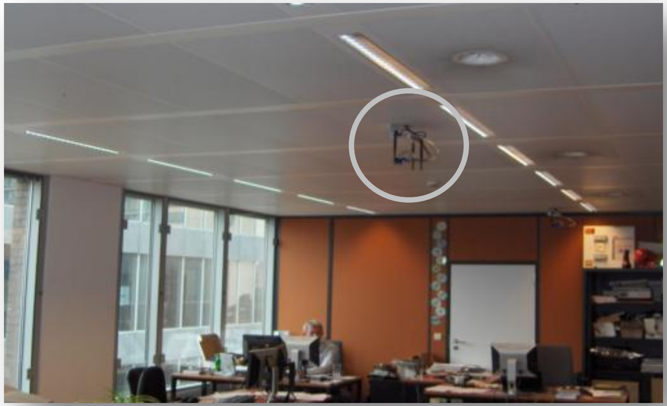
About the iMinds iLab.t

- The iLab.t research centre is located in Ghent

w-iLab.t : Facts and figures

- heterogeneous, generic testbed for wireless networks
  - Sensor networks
  - Wi-Fi based wireless ad-hoc/mesh/vehicular

- 2 testbed locations
  - Office: three office floors of 90m x 18m [200 nodes]
  - “Pseudo-shielded”, Zwijnaarde, 60m x 20m [60 nodes]

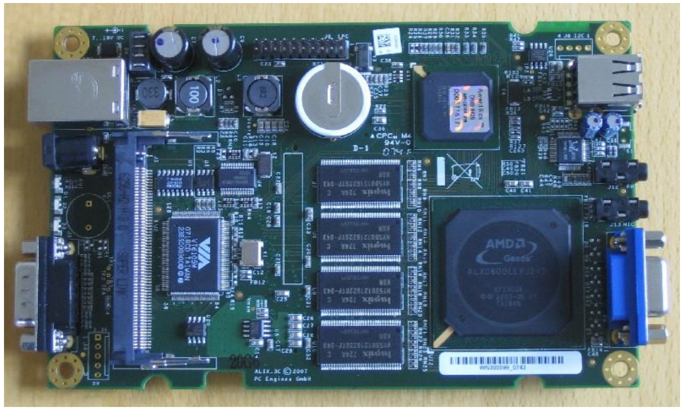


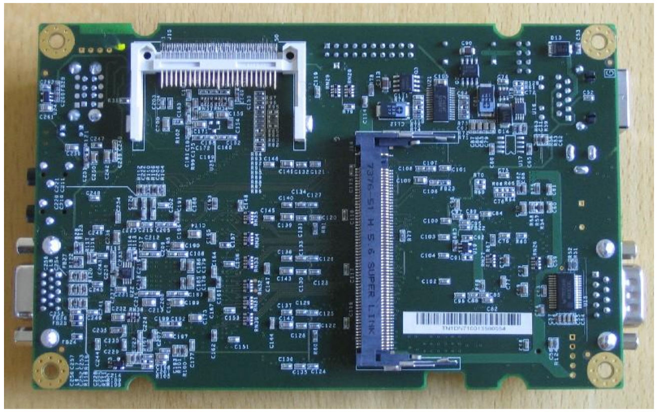
w-iLab.t

Hardware overview

Hardware – Embedded PC

- w-iLab.t Office
  - Alix 3c3 (500 MHz AMD, 256 MB Ram)
    - Ethernet NIC (100Mbit) / Serial port
    - VGA, onboard audio
    - compact flash storage (2 Gb)
    - 2 x mini PCI slot
    - USB

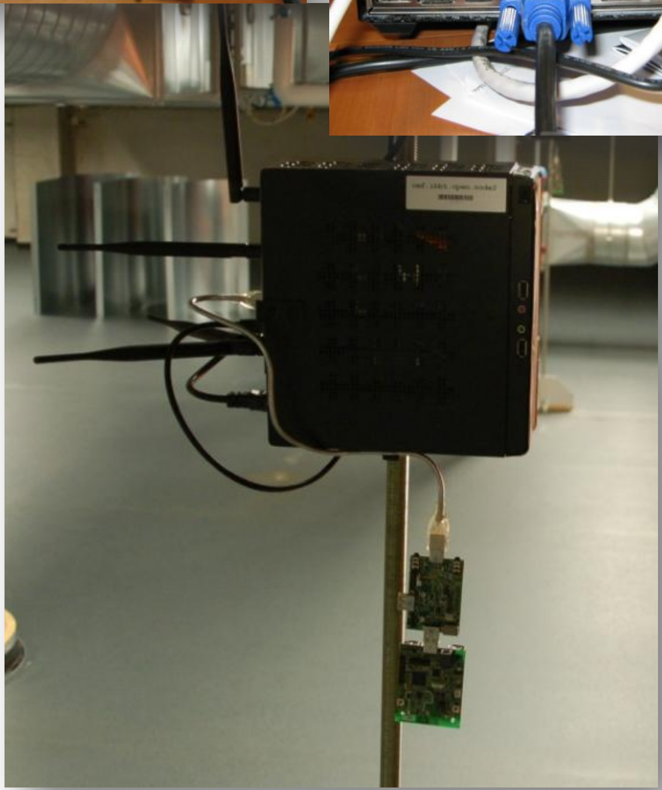




Picture source: www.pcengines.ch

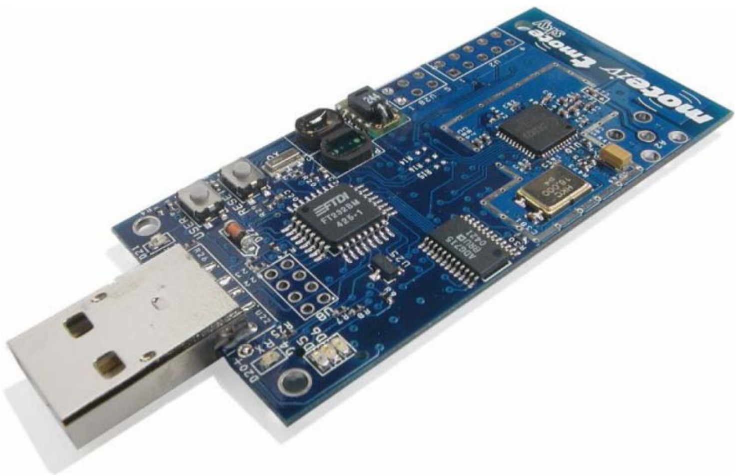
Hardware – Embedded PC

- w-iLab.t Zwijnaarde
  - Zotac
    - 4Gb RAM
    - 160 Gb Hard Drive
    - Intel Atom D510 1.66GHz Dual core
  - Wireless interfaces :
    - 802.11a/b/g/**n** (x2)
    - **Bluetooth**
    - **iMinds Rmoni sensor node** (802.15.4)
  - Environment Emulator
  - **Webcam** (20%)

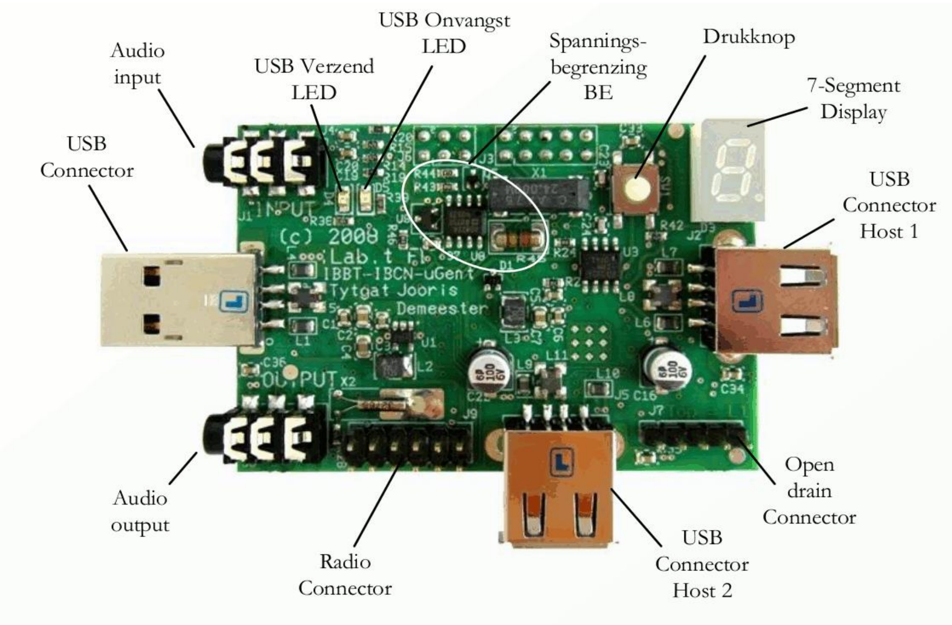


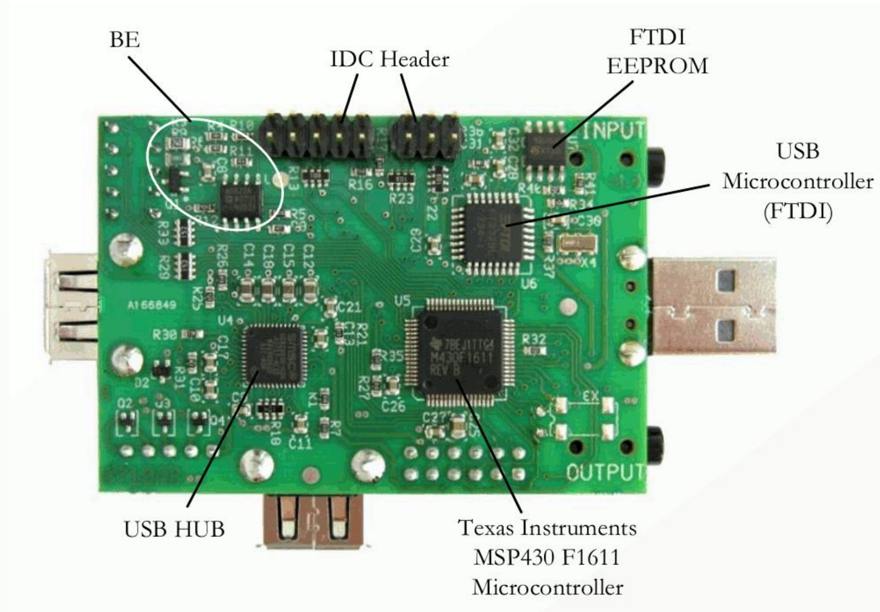
Hardware – sensor devices

- Tmote Sky (office) & iMinds Rmoni (Zwijnaarde)
  - TI msp430
  - CC2420 or CC2520
  - Sensors for temperature and humidity
- Specifications available on www.crew-project.eu/portal



Hardware: environment emulator

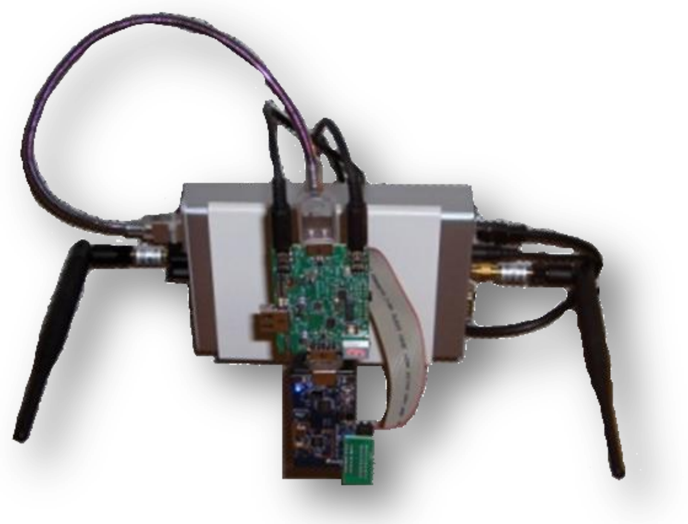




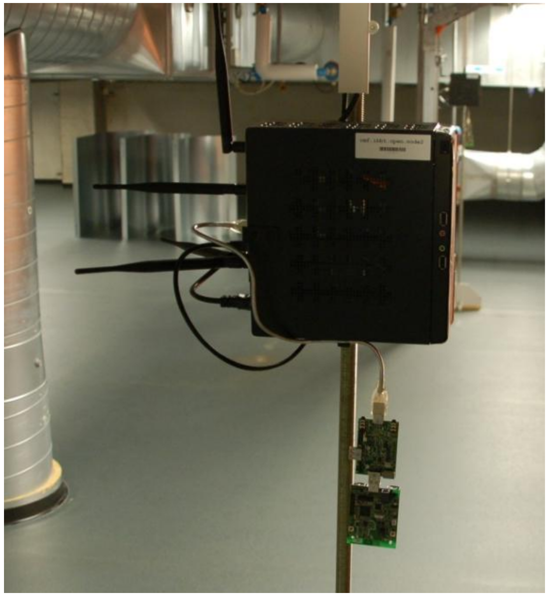
More information: L. Tytgat, B. Jooris, P. De Mil, B. Latré, I. Moerman, P. Demeester, **"Demo abstract: WiLab, a real-life wireless sensor testbed with environment emulation"**, published in European conference on Wireless Sensor Networks, EWSN adjunct poster proceedings (EWSN), Cork, Ireland, 11-13 February 2009

Node locations

- w-iLab.t Office (x200)
  - Node = Embedded PC + EE + Tmote Sky

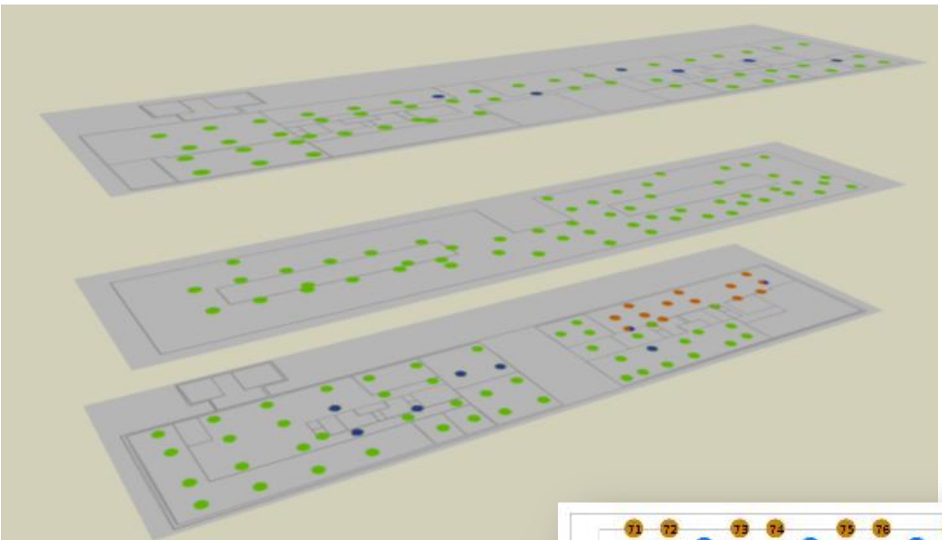


- w-iLab.t Zwijnaarde (x60)
  - Node = Embedded PC + EE + RM090



Topology

- w-iLab.t Office

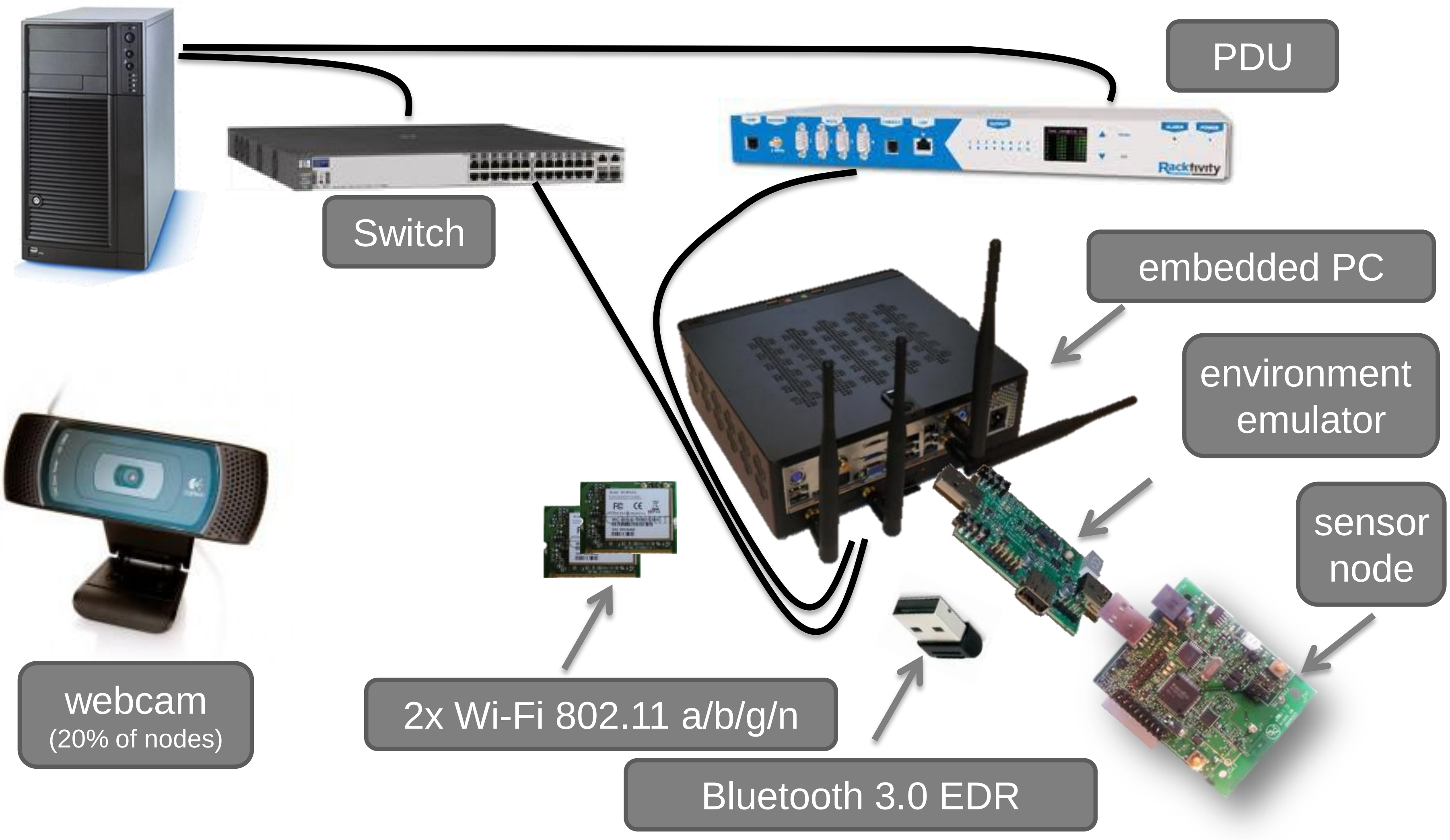


- w-iLab.t Zwijnaarde

w-iLab.t

Testbed Architecture & Management

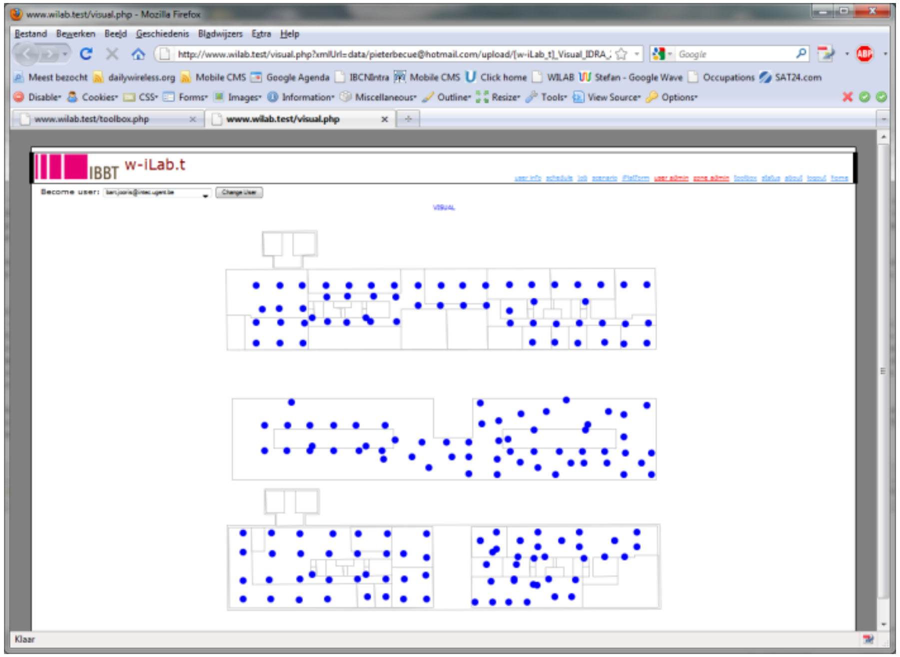
Testbed Architecture

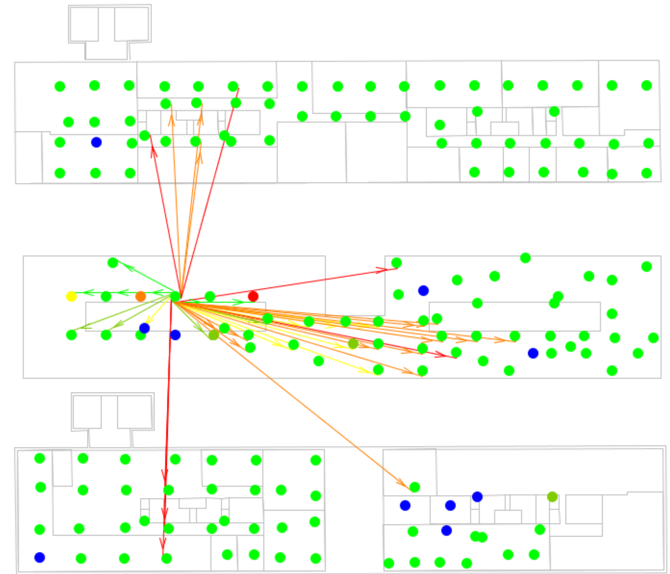


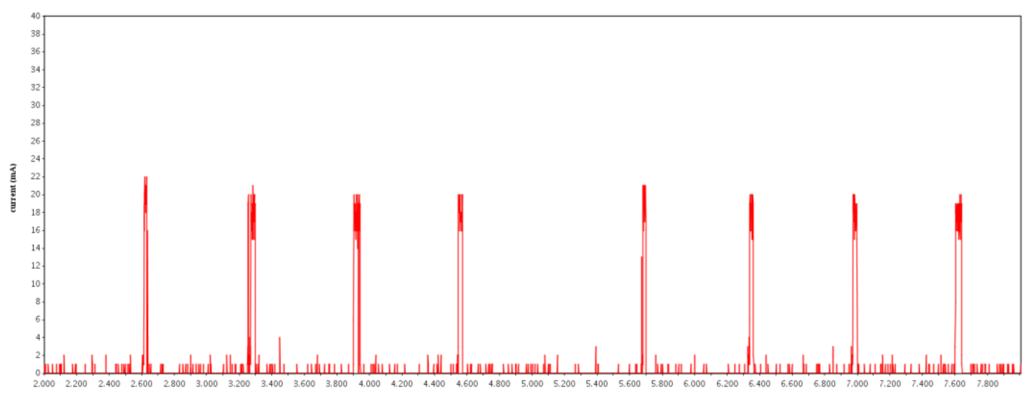
Management Framework & Tools

- OMF (http://omf.mytestbed.net)
- OML for collecting measurements (http://oml.mytestbed.net)

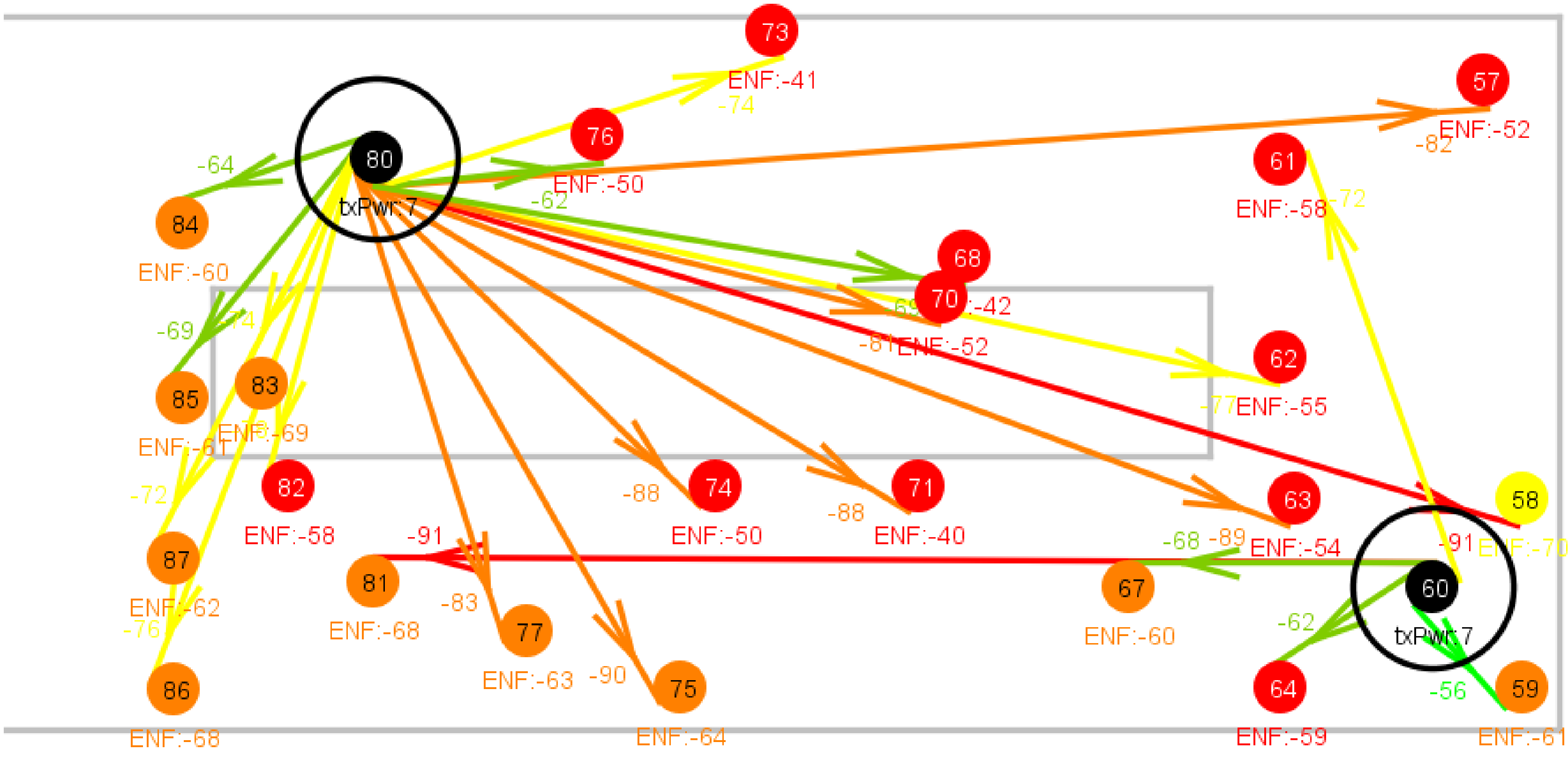
Visualization toolbox







Tx power – RSSI - ENF



w-iLab.t

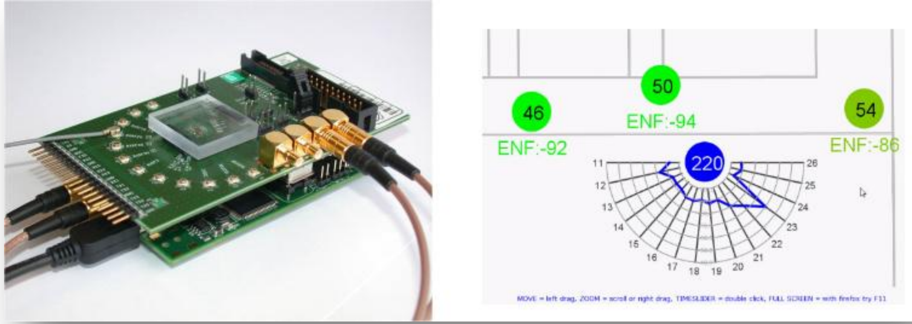
Testbed Hardware Extensions

Shielded testing environment

- shielded from outside interference
- coax connected
- variable attenuators
- emulate mobility

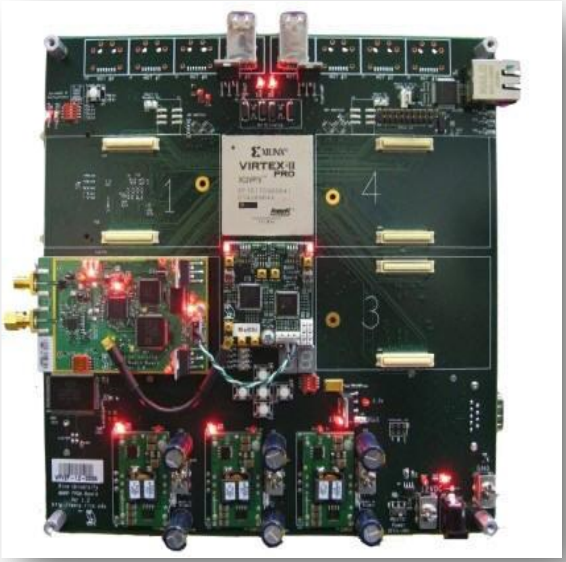
Cognitive components

- imec sensing engine



- USRP2 (SDR radios)

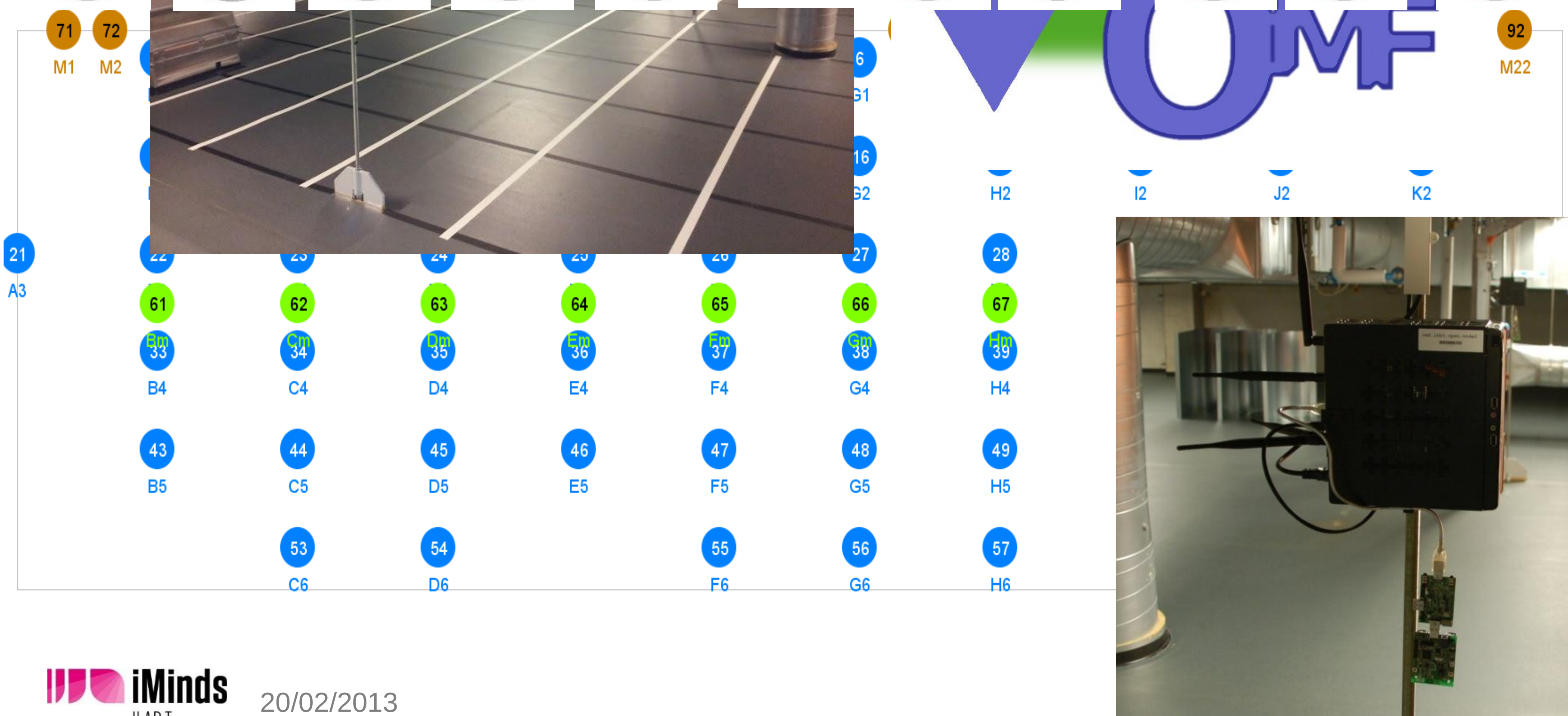
- Rice WARP



Specifications available on
www.crew-project.eu/portal

Mobility @ w-iLab.t

Easy-to-use
Repeatable mobility!



Mobility Extension

- allow experiments with mobile nodes
- based on vacuum cleaning robot
  - extended with:
    - iMinds Robotcontrol
      - In-house designed circuit board (Power control)
      - radio for remote control (ez430 – 868MHz)
    - Embedded PC
      - Powered by external battery pack
      - Webcam
      - Wireless interfaces :
        - 802.11a/b/g/n
        - Bluetooth
        - iMinds Rmoni sensor node (802.15.4)

Robot Demo

1. Robot will drive a rectangular path

2. Toggle its “eye” on a specific coordinate

w-iLab.t

OMF Introduction

OMF Tutorial

Thierry Rakotoarivelo

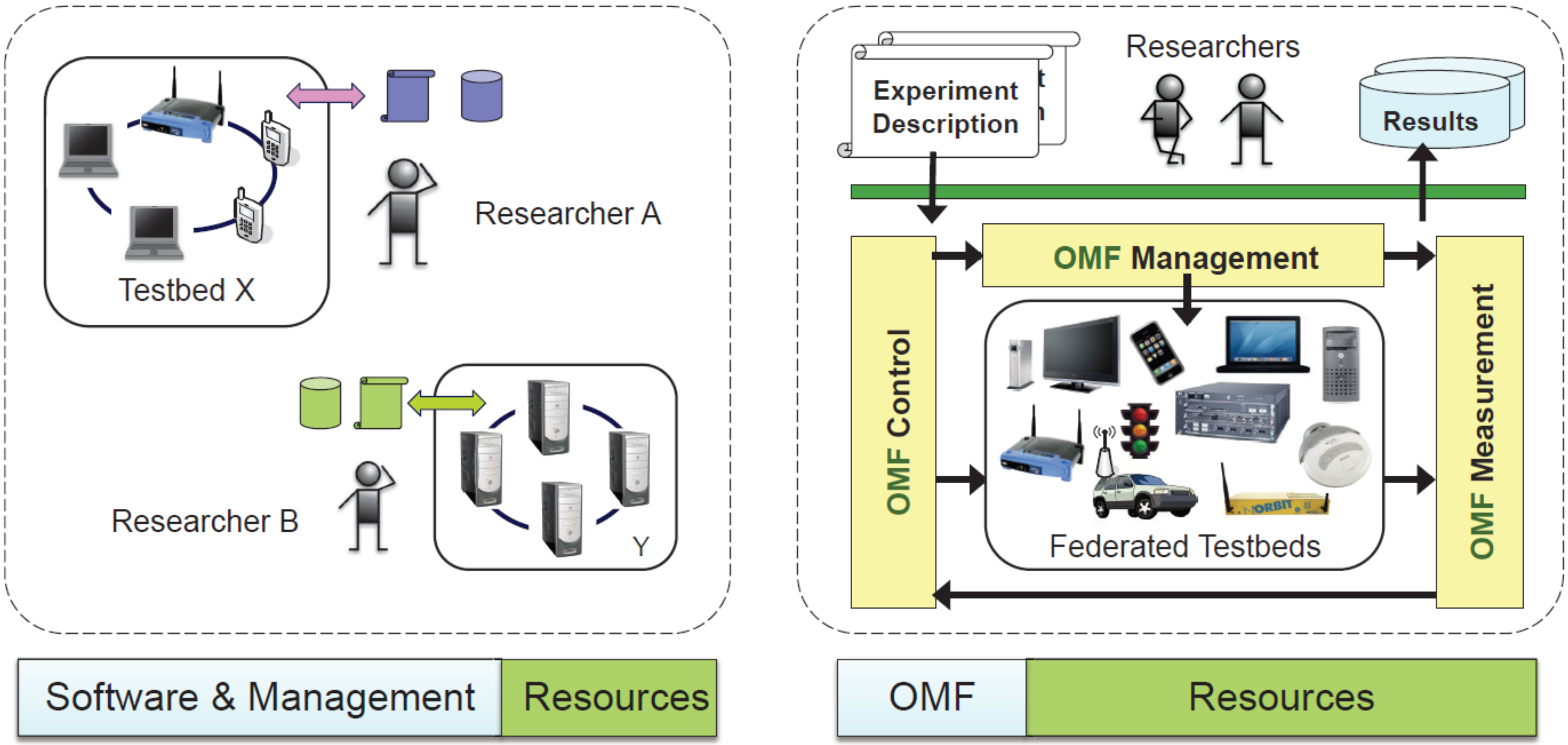
NICTA

Australian Government
Department of Broadband, Communications and the Digital Economy
Australian Research Council

NICTA Funding and Supporting Members and Partners

iMinds
ILAB.T

The Problem and Our approach



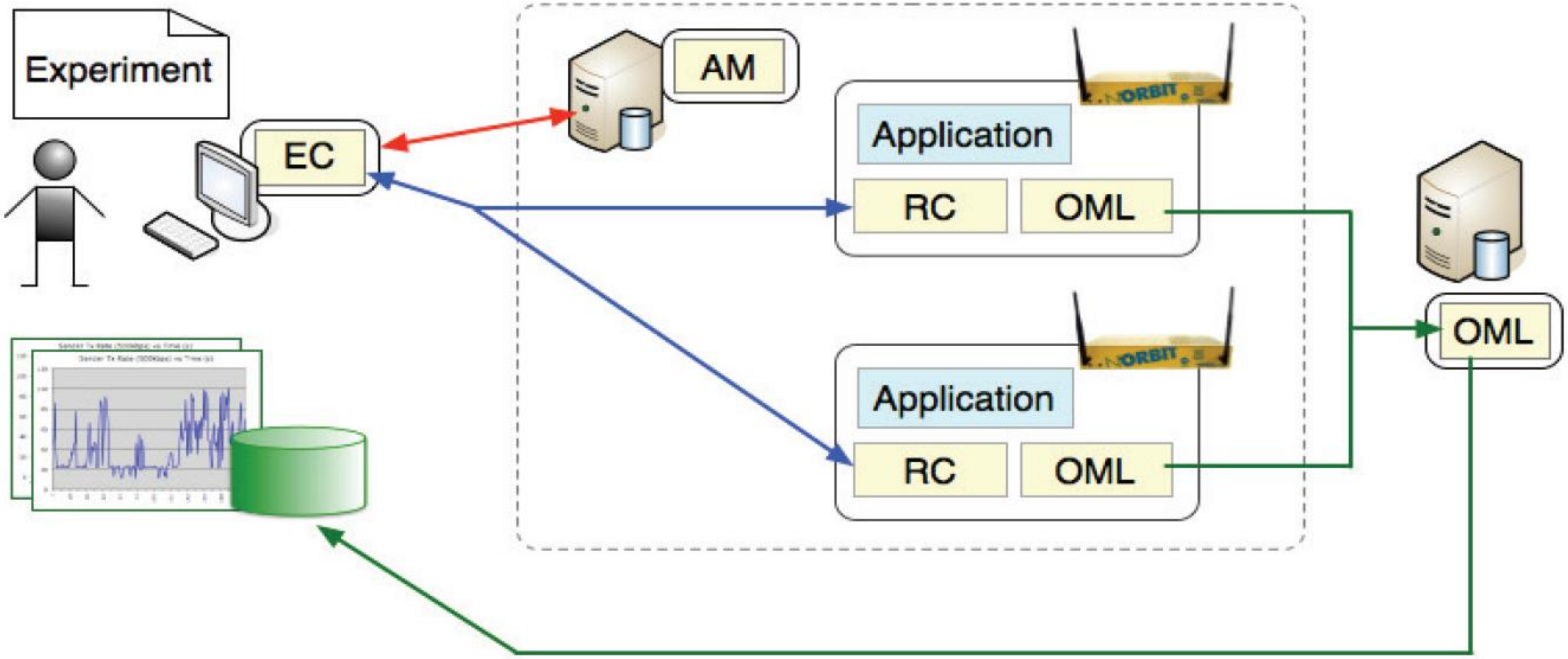
Support & share different resources

Federation of different testbeds

OMF deployment worldwide

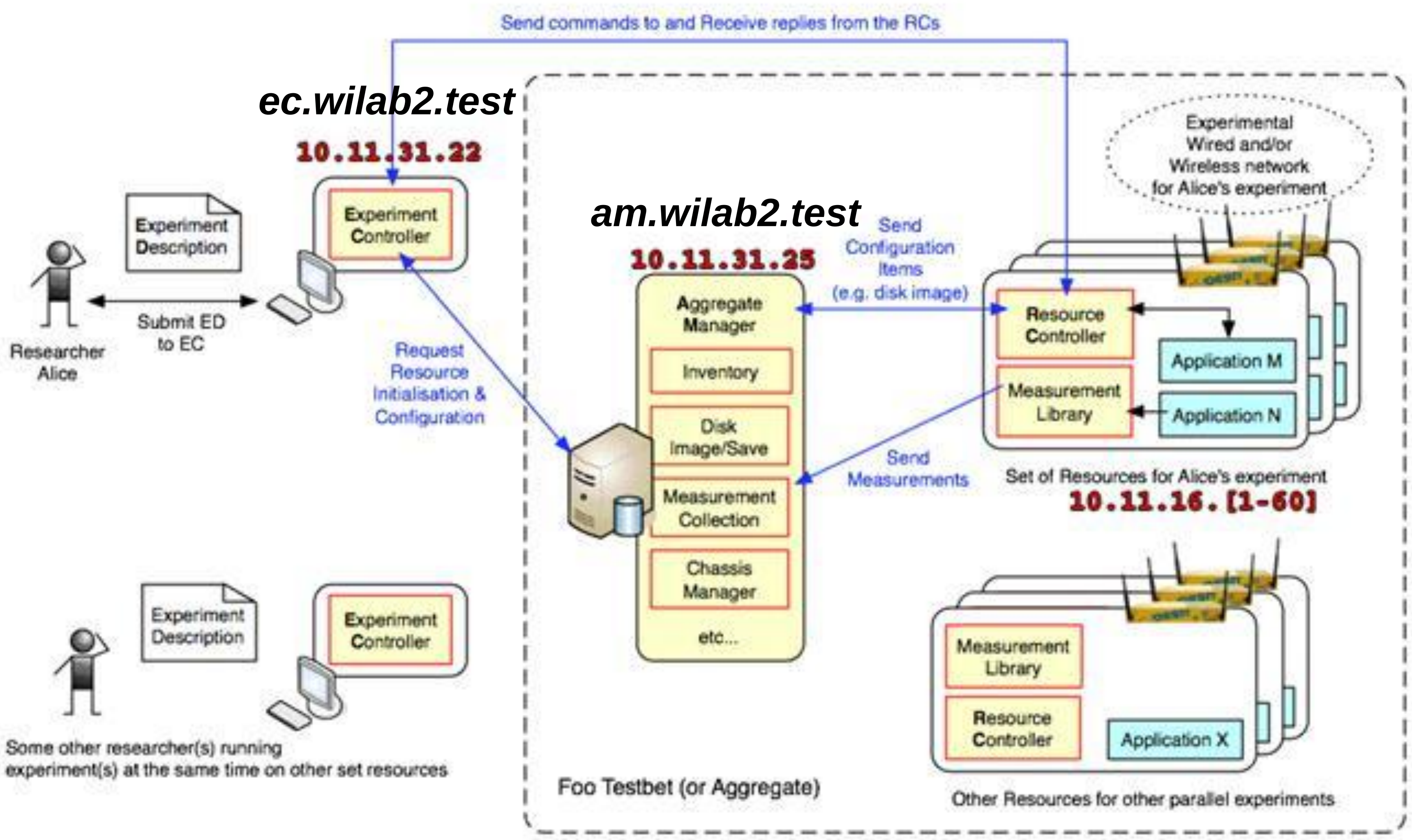
How it works from a user's perspective?

NICTA



iMinds
ILAB.T

OMF Setup @ w-iLab.t Zwijnaarde



Using sensor nodes with OMF / OML

- Some OMF extensions:
  - to flash the sensor node
  - to start “gateway” applications (i.e. Serial Forwarder)
  - to parse output data (and store it in OML)
  - to send input data to the sensor node (through SF)

- The user has to:
  - Provide its own mapping file (packet layout)
  - Include gateway and parsing scripts in the Experiment Description

w-iLab.t

OMF Tutorial – Step by step

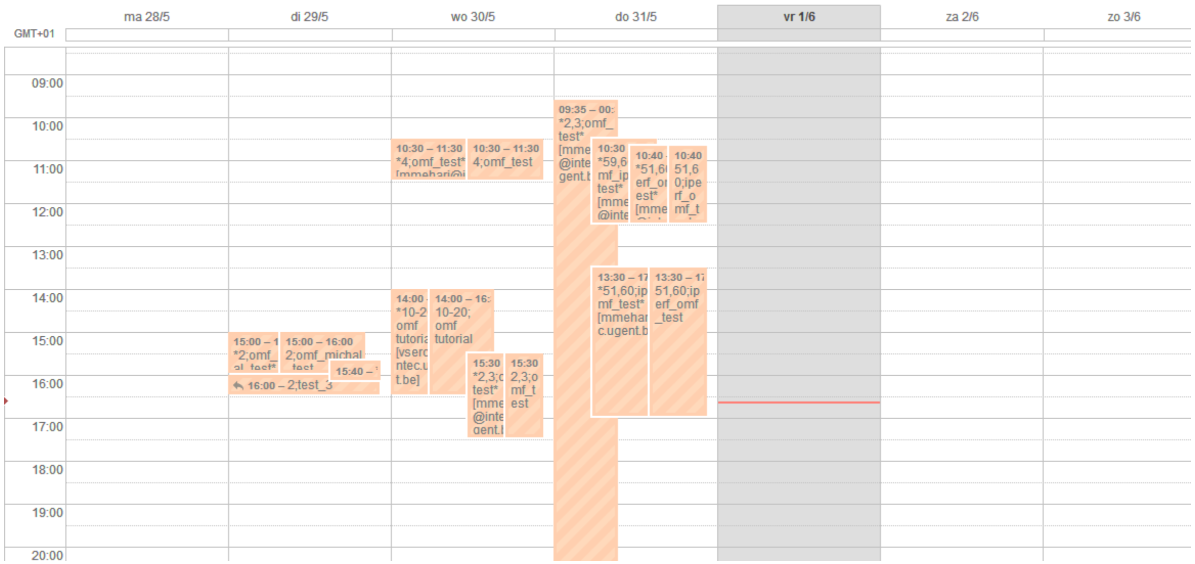
Resource discovery

(make sure your VPN connection is up !)

- w-iLab.t Zwijnaarde webinterface
  - http://wilab2.test/
  - Functionality
    - Nodes availability
    - Node power control
  - Lots of tutorials !

Reservation

- Google Calendar account
  - “Invite ibbtwilab2@gmail.com”
- No real reservation
  - Gentlemen’s agreement
  - Check performed if nodes are still available

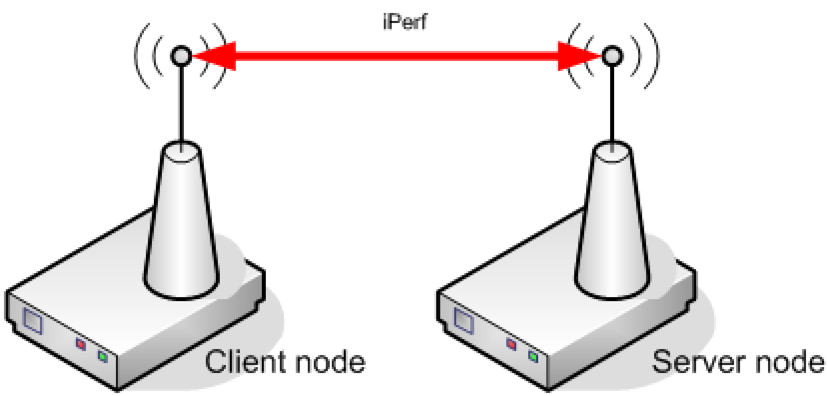


Installation of OS

- Stored in /var/lib/omf-images-5.3 (Experiment Controller)
  - omf load -t omf.ibbt.open.nodeX,omf.ibbt.open.nodeY -i imagename.ndz
- Image is multicasted
  - Loading time independent of nr of nodes
- iMindsBaseline.ndz (login: root/urbis)
  - Ubuntu 12.04
  - Sensor tools (communication and flashing tools)
  - Webcam tools
  - Bluetooth tools

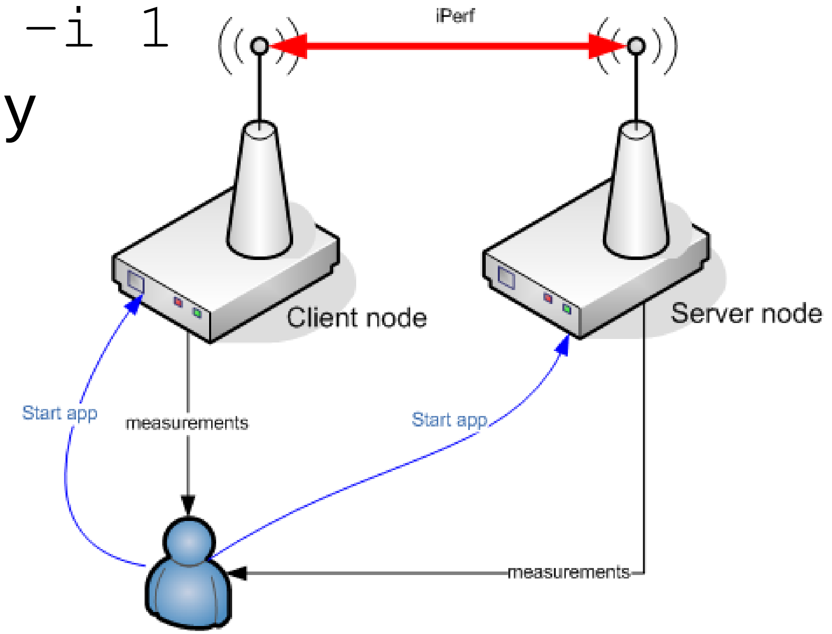
Running basic OMF Experiments: demo case

Goal: Test throughput of Wi-Fi using iPerf



Running basic OMF Experiments: manual approach

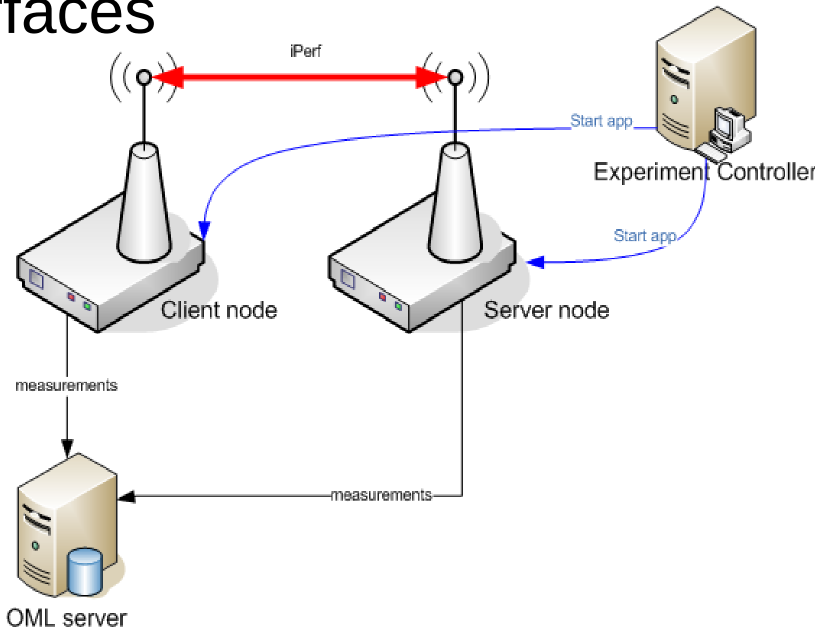
1. ssh to the server
2. set up wireless interface
3. start serverside application with `iperf -s -u`
4. ssh to the client
5. set up wireless interface and connect it to the server
6. start clientside application with: `iperf -c [serverip] -u -b 1000M -i 1`
7. collect measurements manually



iMinds
ILAB.T

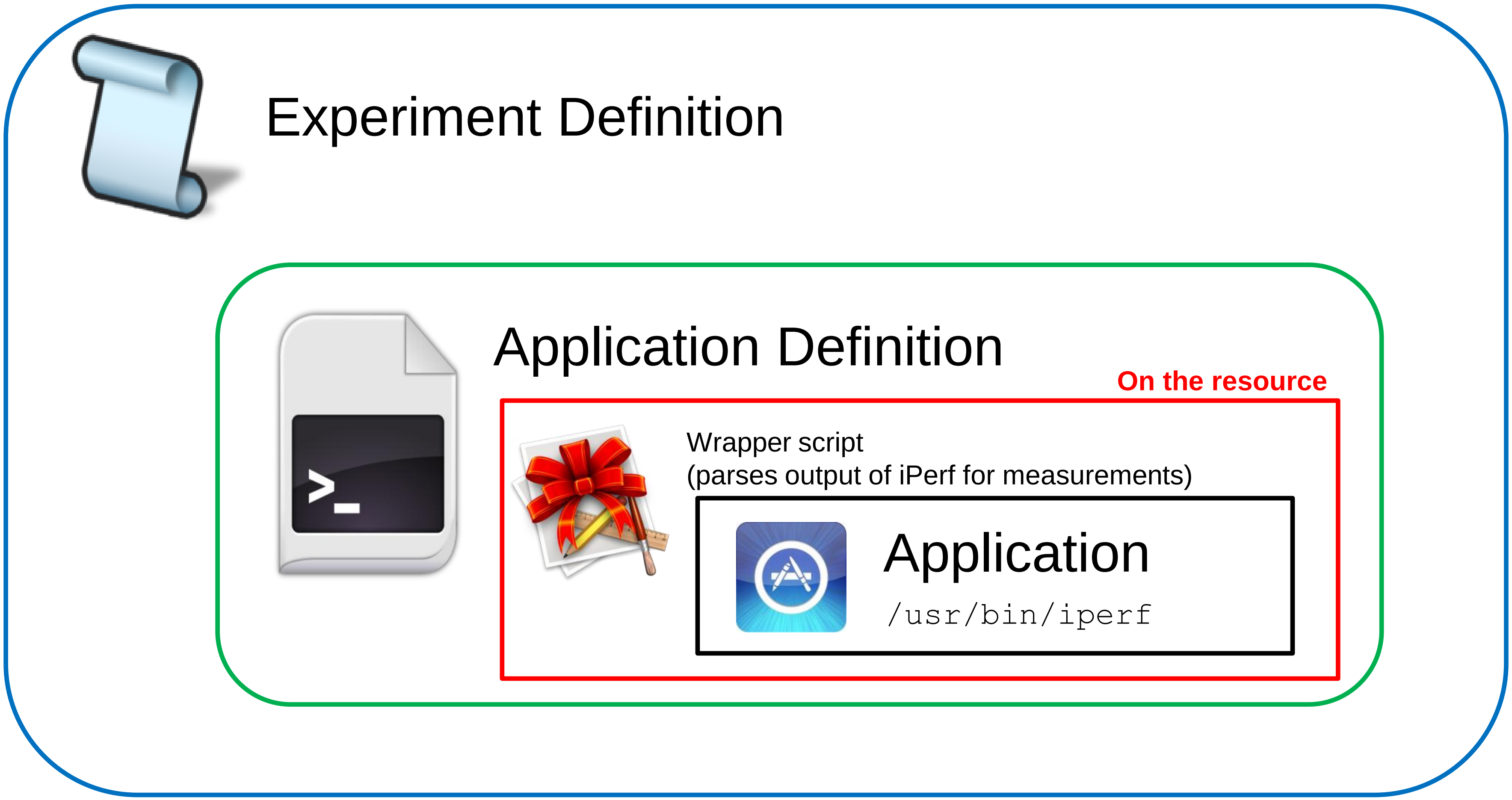
Running basic OMF Experiments: the OMF way

1. ssh to the experiment controller
2. write an **application definition**
   - where can OMF find iPerf?
   - what are valid commandline arguments?
   - what is the output of iPerf?
3. write an **experiment description**
   - configuration of wireless interfaces
   - which nodes?
   - timeline of the experiment
4. execute the experiment
5. read measurements in database



iMinds
ILAB.T

Running basic OMF Experiments



Running basic OMF Experiments

Experiment Definition

```
defGroup("server","omf.ibbt.open.node1")
    <config wireless interface>
    <add application(s)>
defGroup("client","omf.ibbt.open.node3")
    <config wireless interface>
    <add applications(s)>
```

```
defApplication('iperfwrap', 'iperfwrap')
```

```
onEvent(:ALL_UP_AND_INSTALLED) do |event|
  info "iPerf experiment - wait for interfaces to come up"
  wait 5
  group("server").startApplications
  info "Starting stream 1 server..."
  wait 2
  info "Starting stream 1 client..."
  group("client").startApplications
  wait 60
  group("client").stopApplications
  group("server").stopApplications
  Experiment.done
end
```

Application Definition

Wrapper script
(parses output of iPerf for measurements)

Application

`/usr/bin/iperf`

Running basic OMF Experiments

Experiment Definition

Application Definition

```
defApplication('iperfwrap', 'iperfwrap') do |app|
  app.path = File.join(File.dirname(__FILE__), 'iperfwrap.rb')
  app.appPackage = "iperfwrap.tar"

  app.shortDescription = "iPerf wrapper"
  app.description = "Simple iPerf wrapper with OML logging"

  app.defProperty('server', 'Act as server (true/false)', 's' , ...)
<more properties>

  # the measurement definition
  app.defMeasurement('iperfmp') do |mp|
<metrics: i.e. the fields in every record in the database>
  end
end
```

Wrapper script
(parses output of iPerf for measurements)

Application

`/usr/bin/iperf`

Running basic OMF Experiments

Experiment Definition

Application Definition

Wrapper script

(parses output of iPerf for measurements)

```
require "/usr/share/liboml2-dev/oml4r.rb"

APPPATH = "/usr/bin/iperf`

class MPStat < OML4R::MPBase
        name :iperfmp
        param :hostname
        param :interval_begin, :type => :double #sec
        param :interval_end, :type => :double #sec
        <other metric definitions>
end

<logic to pass commandline args to iperf>
<logic to parse output and generate OML>
```

Application

`/usr/bin/iperf`

Run it!

- ssh to ec.wilab2.test as user: **`demoX`** pass: **`demoX`**
- navigate to **`~/ewsn/iperf`**
- inspect **`experiment_description.rb`**
- inspect **`iperf_wrapper.rb`**
- run the experiment using
  `omf exec experiment_description.rb`
- Log in on a node and execute
  `ps -aux | grep iperf`
- paste your experiment ID
  in the textbox on http://wilab2.test/
  measurements tab

	nodeA	nodeB
reserved	25	26
demo1	2	3
demo2	4	5
demo3	6	7
demo4	8	9
demo5	13	14
demo6	15	16
demo7	17	18
demo8	19	20
demo9	22	23
demo10	27	28
demo11	29	30
demo12	33	34

The w-iLab.t testbed

details:
http://ilabt.iminds.be/

vincent.sercu@intec.ugent.be
pieter.becue@intec.ugent.be
bart.jooris@intec.ugent.be
stefan.bouckaert@iminds.be
www.ibcn.intec.ugent.be – www.iminds.be