

**NITOS TESTBED:
A RESEARCH AND EXPERIMENTATION
FACILITY FOR IMPLEMENTING AND
EVALUATING WIRELESS NETWORKS**

**THANASIS KORAKIS
ITI-EKETA**

PART 1

NITOS PRESENTATION

Part 1 – Overview

- ▣ **NITOS Description**
- ▣ **How to use NITOS Testbed**
- ▣ **NITOS Management System – Scheduler**
- ▣ **NITOS Management System – Accessing the Network**
- ▣ **NITOS Management System – OMF**

NITOS testbed status

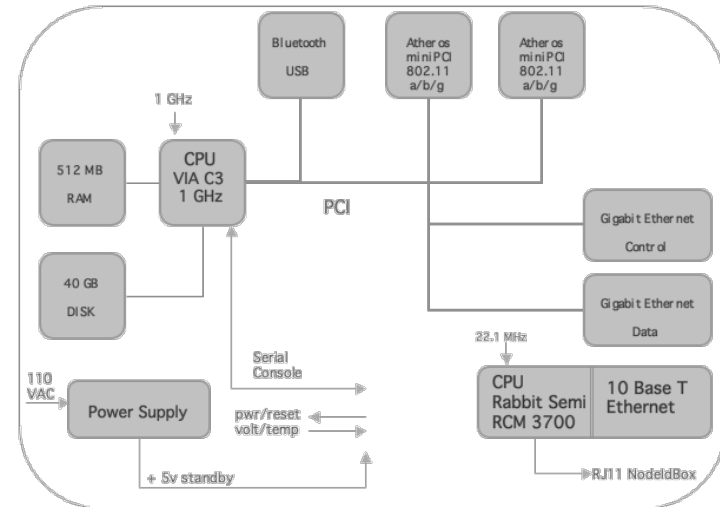
- Currently NITOS maintains
 - 40 Wi-Fi Nodes**
 - 6 of them are equipped with GNU radios**
 - 15 of them are equipped with video cameras**
 - 10 of them are equipped with temperature/humidity sensors**

ORBIT-like Nodes



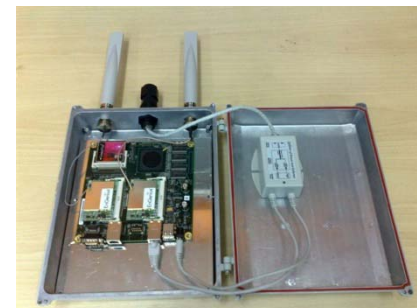
- **Technical Specifications**

- 1 GHz VIA C3 processor
- 1GB of RAM
- 40 GB disk
- two ethernet ports
- two 802.11 a/b/g cards
- a Chassis Manager

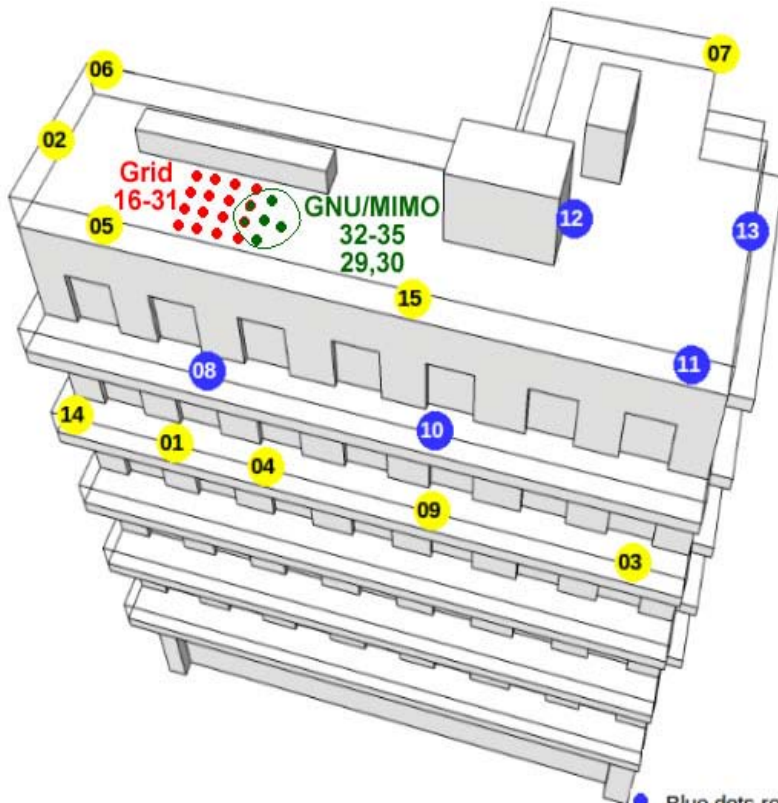


Diskless Nodes

- **ALIX2.C2 boards PC ENGINES**
 - Lack of hard disk
 - Inexpensive compared to Orbit nodes
- **Technical Specifications**
 - CPU: 500 MHz AMD Geode LX800
 - DRAM: 256 MB DDR DRAM
 - Storage: Compact Flash socket
 - Power: DC jack or passive POE, min. 7V to max. 20V
 - Expansion: 2 mini PCI slots, LPC bus
 - Connectivity: 2 Ethernet channels (Via VT6105M 10/100)



Testbed Nodes Topology



- Grid nodes 4x4
- Orbit like nodes
- Diskless nodes
- 6 Nodes with GNU and MIMO radios

• Blue dots represent Diskless Nodes
• Yellow dots represent Orbit Nodes
• Red dots represent Commell Nodes
• Green dots represent GNU/MIMO Nodes

Cameras

15 of the wireless nodes are equipped with Logitech C120 USB web-cameras.



- NITOS aims to provide a Video Framework to support:
 - Monitoring and capturing real time video streams
 - QoS for video streaming in wireless networks
 - Cooperation protocol for reliable multicast video streams
 - Easy configuration of coding/decoding transmitted frames

GNU Radios

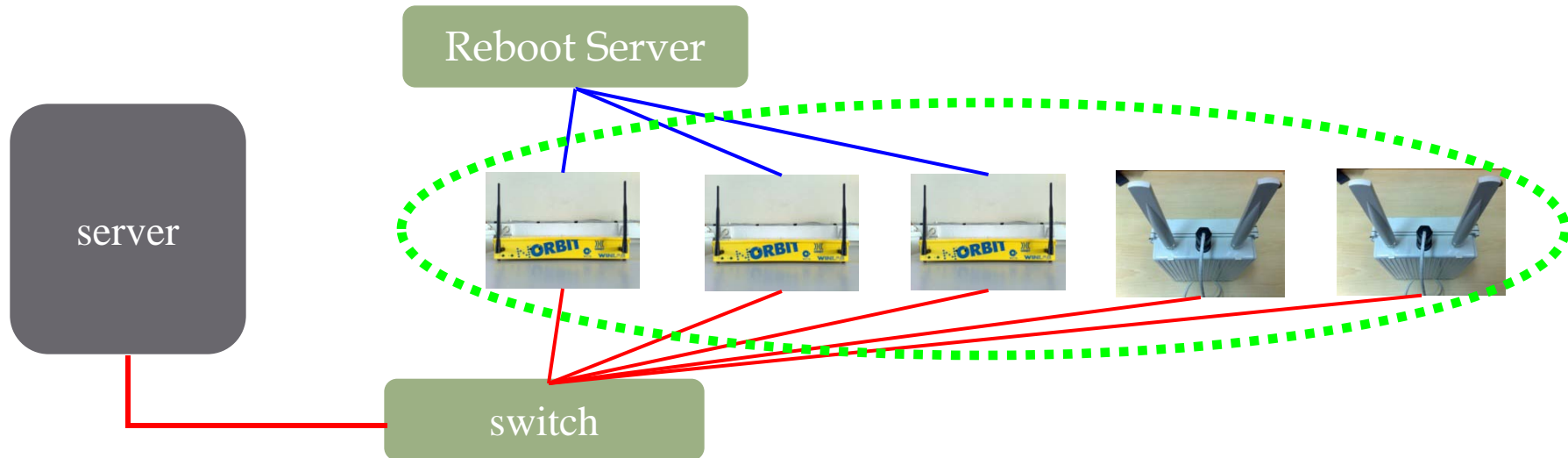
6 of the nodes are connected with GNU Radio boards.

- NITOS aims to provide a Framework to support:
 - Physical and MAC layer implementation
 - OMF for better integration of GNU radio software
 - Subcarrier allocation
 - Power control



Testbed Network Overview

- **Reboot Server**(AC power cables)
- **Experiment Network** (Ethernet)
- **Experiment Network** (Wireless)



NITOS Testbed

-

How to use it

NITOS web site

- <http://nitlab.inf.uth.gr/NITlab/>

Monday, September 06, 2010 Login

NITlab
NETWORK IMPLEMENTATION TESTBED LABORATORY

Nitlab Search
Search...

Home Testbed Scheduler Projects News Contact us

Temperature & Humidity Sensors in NITOS

Usb THUM (Temperature and Humidity Monitor) Sensors are now connected on the testbed's nodes!

EVENTS

- ▶ **WINTech 2010**
Dr. Thanasis Korakis, Associate Director of NITlab, is Program Chair of WINTech 2010, a major workshop of MobiCom, focussing on...
- ▶ **New Reboot Server**
The new reboot server is ready for integration with NITOS.
- ▶ **Nodes are ready**
The new nodes are ready for deployment! The NITlab staff starts the deployment of the new nodes! The new Testbed will count 40 nodes...

NITLAB - NETWORK IMPLEMENTATION TESTBED LABORATORY

Network Implementation Testbed Laboratory

Welcome to the Network Implementation Testbed Laboratory of the **Computer and Communication Engineering Department** at **University of Thessaly**. NITLab is also affiliated with the **Centre for Research & Technology Hellas (CERTH)**. The research of the lab focuses on the design, study and implementation of wireless schemes and their performance in the real environment. In this context, NITLab has developed a testbed named **NITOS**, which stands for **Network Implementation Testbed** using **Open Source** code. NITOS is a wireless testbed that consists of nodes based on open source software.


The **NITOS** platform is open to any researchers who would like to test their protocols in a real-time wireless

MAIN MENU

- ▶ Home
- About
- People
- Publications
- ▶ Testbed

NITOS registration

Monday, September 06, 2010



Home Testbed Scheduler Projects News Contact us

REGISTRATION

Fields marked with an asterisk (*) are required.

Name: *

Username: *

E-mail: *

Password: *

Verify Password: *

EVENT

- ▶ Win...
Dr. T...
NITla...
2010...
focu...
- ▶ Nev...
The...
integ...
- ▶ No...
The...
The...
40 n...

MAIN

▶ Ho...

NITOS Management System - Scheduler

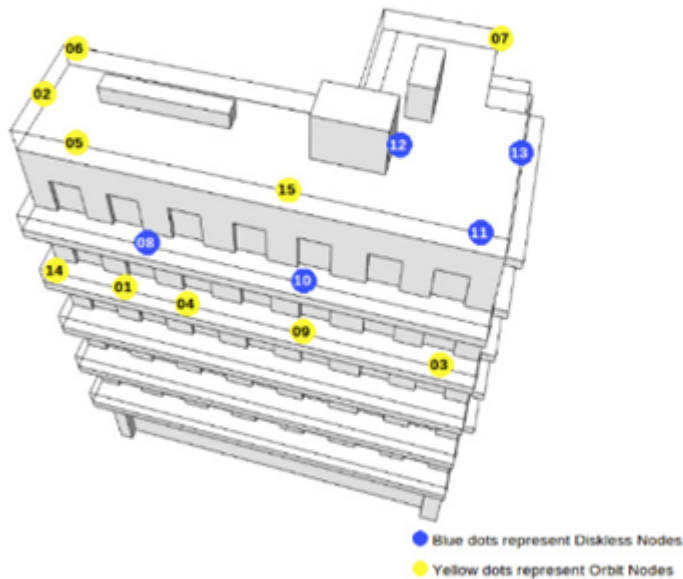
Scheduler Components

- ▣ The NITOS Scheduler consists of:
 - A database where information about the testbed's availability is kept.
 - A web interface which provides experimenters the necessary information and tools to make and watch their reservations.
 - A web-embedded application which navigates the users in the testbed and allows them to reserve a slice at the time they would like.
 - A set of server-side scripts responsible for communication between the client and the database mentioned above.

NITOS Reservation Procedure

- First the user chooses the date that he would like to use the testbed:

Welcome to the NITOS Testbed Reservation Tool



Current Server Time: 2010-05-24 20:48:00

Click on a date to select the day you want to start the reservation.

<< May 2010 >>						
Sun	Mon	Tue	Wen	Thu	Fri	Sat
						01
02	03	04	05	06	07	08
09	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

Choose the time you want to start the reservation and for how long (max 4 hours).

Select Start Time and Duration

Start Date: yyyy/mm/dd

Start Time: 00 hh 00 min

Duration: 0.5 hours

Check Available Nodes

NITOS Reservation Procedure

- Then the user gets the available nodes for that date.

Available Nodes and Channels between 2010-05-31 15:00:00 and 2010-05-31 18:00:00 :

Nodes	802.11a	802.11b/g	
Node 1: <input type="checkbox"/>	Channel 36: <input type="checkbox"/>		<input type="button" value="Reserve"/>
Node 2: <input type="checkbox"/>	Channel 40: <input type="checkbox"/>	Channel 1: <input type="checkbox"/>	
Node 3: <input type="checkbox"/>	Channel 44: <input type="checkbox"/>	Channel 2: <input type="checkbox"/>	
Node 4: <input type="checkbox"/>	Channel 48: <input type="checkbox"/>	Channel 3: <input type="checkbox"/>	
Node 5: <input type="checkbox"/>	Channel 52: <input type="checkbox"/>	Channel 4: <input type="checkbox"/>	
Node 6: <input type="checkbox"/>	Channel 56: <input type="checkbox"/>	Channel 5: <input type="checkbox"/>	
Node 7: <input type="checkbox"/>	Channel 60: <input type="checkbox"/>	Channel 6: <input type="checkbox"/>	
Node 8: <input type="checkbox"/>	Channel 64: <input type="checkbox"/>	Channel 7: <input type="checkbox"/>	
Node 9: <input type="checkbox"/>	Channel 100: <input type="checkbox"/>	Channel 8: <input type="checkbox"/>	
Node 10: <input type="checkbox"/>	Channel 104: <input type="checkbox"/>	Channel 9: <input type="checkbox"/>	
Node 11: <input type="checkbox"/>	Channel 108: <input type="checkbox"/>	Channel 10: <input type="checkbox"/>	
Node 12: <input type="checkbox"/>	Channel 112: <input type="checkbox"/>	Channel 11: <input type="checkbox"/>	
Node 13: <input type="checkbox"/>	Channel 116: <input type="checkbox"/>	Channel 12: <input type="checkbox"/>	
Node 14: <input type="checkbox"/>	Channel 120: <input type="checkbox"/>	Channel 13: <input type="checkbox"/>	
Node 15: <input type="checkbox"/>	Channel 124: <input type="checkbox"/>		
	Channel 128: <input type="checkbox"/>		
	Channel 132: <input type="checkbox"/>		
	Channel 136: <input type="checkbox"/>		
	Channel 140: <input type="checkbox"/>		

NITOS Reservation Procedure

- Then the user checks the nodes and the frequencies he would like:

Available Nodes and Channels between 2010-05-31 15:00:00 and 2010-05-31 18:00:00 :

Nodes	802.11a	802.11b/g	
	Channel 36: <input type="checkbox"/>		
	Channel 40: <input type="checkbox"/>		
Node 1: <input type="checkbox"/>	Channel 44: <input type="checkbox"/>		
Node 2: <input type="checkbox"/>	Channel 48: <input type="checkbox"/>	Channel 1: <input type="checkbox"/>	
Node 3: <input checked="" type="checkbox"/>	Channel 52: <input type="checkbox"/>	Channel 2: <input type="checkbox"/>	
Node 4: <input type="checkbox"/>	Channel 56: <input type="checkbox"/>	Channel 3: <input checked="" type="checkbox"/>	
Node 5: <input type="checkbox"/>	Channel 60: <input type="checkbox"/>	Channel 4: <input type="checkbox"/>	
Node 6: <input checked="" type="checkbox"/>	Channel 64: <input checked="" type="checkbox"/>	Channel 5: <input type="checkbox"/>	
Node 7: <input checked="" type="checkbox"/>	Channel 100: <input type="checkbox"/>	Channel 6: <input type="checkbox"/>	
Node 8: <input type="checkbox"/>	Channel 104: <input type="checkbox"/>	Channel 7: <input type="checkbox"/>	<input type="button" value="Reserve"/>
Node 9: <input checked="" type="checkbox"/>	Channel 108: <input type="checkbox"/>	Channel 8: <input checked="" type="checkbox"/>	
Node 10: <input type="checkbox"/>	Channel 112: <input type="checkbox"/>	Channel 9: <input type="checkbox"/>	
Node 11: <input checked="" type="checkbox"/>	Channel 116: <input type="checkbox"/>	Channel 10: <input type="checkbox"/>	
Node 12: <input type="checkbox"/>	Channel 120: <input checked="" type="checkbox"/>	Channel 11: <input type="checkbox"/>	
Node 13: <input type="checkbox"/>	Channel 124: <input type="checkbox"/>	Channel 12: <input type="checkbox"/>	
Node 14: <input checked="" type="checkbox"/>	Channel 128: <input type="checkbox"/>	Channel 13: <input type="checkbox"/>	
Node 15: <input type="checkbox"/>	Channel 132: <input type="checkbox"/>		
	Channel 136: <input type="checkbox"/>		
	Channel 140: <input type="checkbox"/>		

Slicing Support

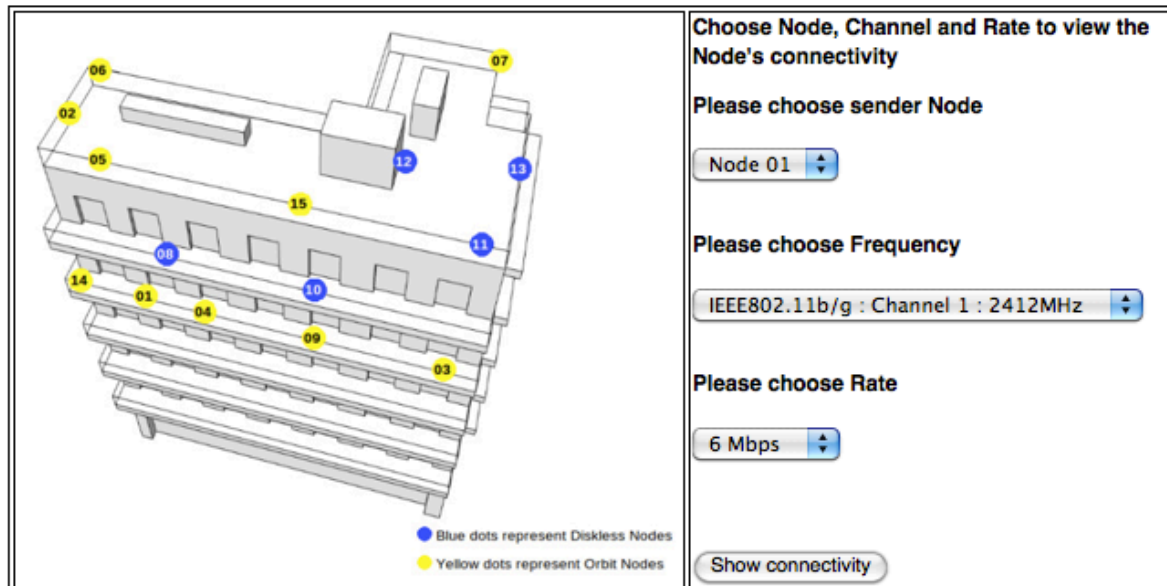
- ▣ NITOS is supporting **spectrum slicing**.
- ▣ Each user chooses the channels he intends to use and as a result, many users may use the testbed **concurrently** without any interference.
- ▣ The Scheduler will prevent from any overlapping taking place on nodes or channels selection.
- ▣ Slicing control is achieved in two ways:
 - Using **CRDA** which works with ath5k driver and restricts the channels used by setting a regulatory domain for each user based on his selection.
 - Using **OMF** which asks the scheduler if it can set the requested channel to the respective resource (i.e. if the user running the experiment has actually reserved that channel).

NITOS Connectivity Tool

- NITOS has developed a connectivity tool which provides information about link quality among nodes
- The data is available on the NITOS website and is updated frequently.

Topology-Connectivity

This is the topology of our testbed residing on the building of Computers and Communications Engineering Department of University of Thessaly in Volos.



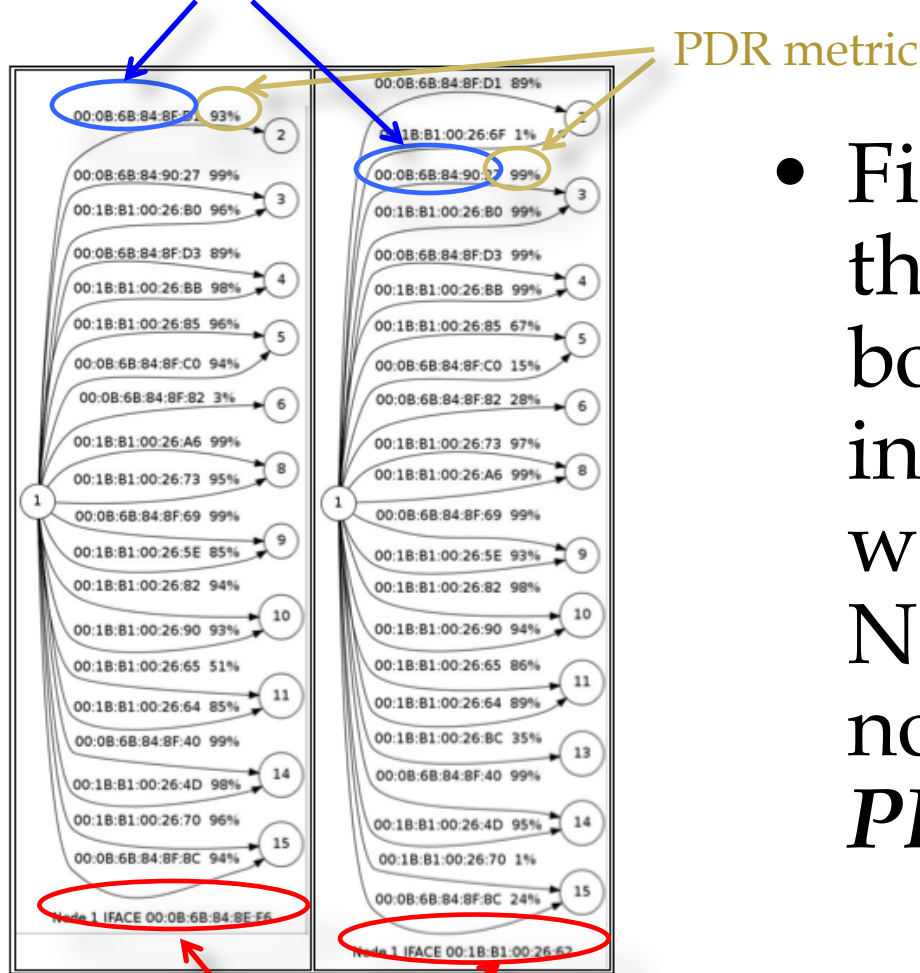
The screenshot displays the NITOS Connectivity Tool interface. On the left, a 3D perspective view of a server rack is shown with 15 nodes. Nodes 01, 02, 03, 04, 05, 06, 07, 09, 10, 11, 12, 13, 14, and 15 are marked with yellow dots, representing Orbit Nodes. Nodes 08 and 15 are marked with blue dots, representing Diskless Nodes. A legend at the bottom of the diagram indicates: "Blue dots represent Diskless Nodes" and "Yellow dots represent Orbit Nodes".

On the right, the control panel includes the following elements:

- Choose Node, Channel and Rate to view the Node's connectivity**
- Please choose sender Node**: A dropdown menu showing "Node 01".
- Please choose Frequency**: A dropdown menu showing "IEEE802.11b/g : Channel 1 : 2412MHz".
- Please choose Rate**: A dropdown menu showing "6 Mbps".
- Show connectivity**: A button to execute the connectivity check.

NITOS Connectivity Tool

Receiver's MAC address



PDR metric

- Figure illustrates the connectivity of both wireless interfaces of a node with all the other NITOS testbed nodes, based on *PDR* metric

Sender's MAC address

Connectivity of Node 1 operating on Channel 1 at Rate 6 Mbps.

NITOS Management System

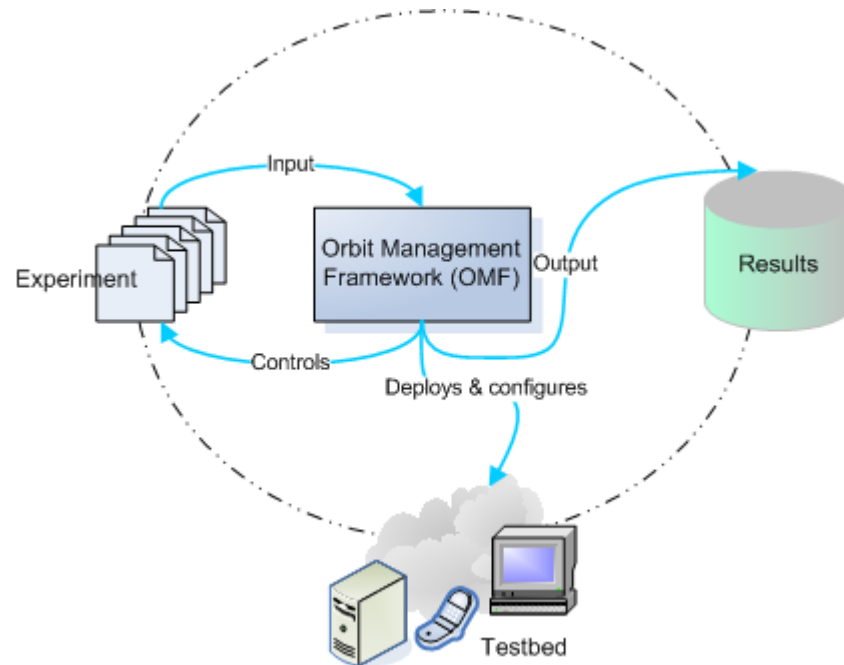
-

OMF

<http://www.omf.mytestbed.net/>

OMF Workflow

- ▣ The user submits his experiment to OMF.
- ▣ OMF configures the testbed and begins the experiment execution.
- ▣ OMF collects the experiment results in a database.



PART 2

RESEARCH ACTIVITIES ON NITOS

Part 2 – Overview

Experimentation with NITOS:

- ▣ **MAC layer implementation activities**
- ▣ **Network layer implementation activities**
- ▣ **Other implementation activities**
 - **Video**
 - **Power meter**
 - **Sensors**

MAC LAYER RESEARCH ACTIVITIES

Dynamic Frequency Selection

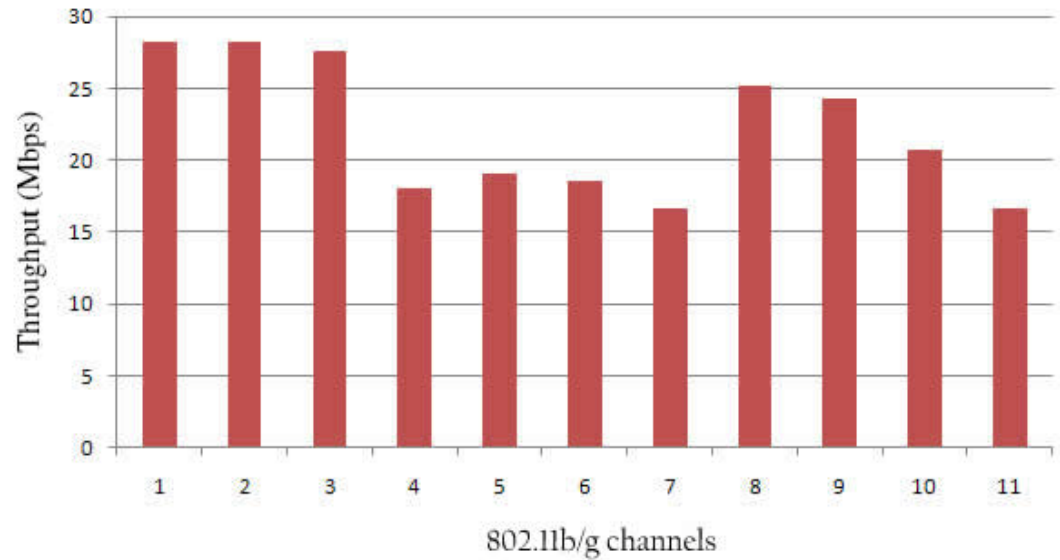
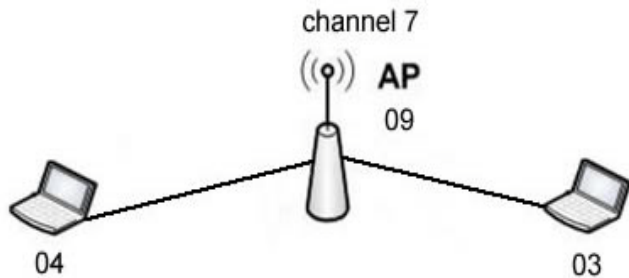
DFS

- ▣ **DFS is used to :**
 - ✓ **switch the operational frequency channel of a BSS to another dynamically , in order to avoid interference with other devices, such as radar systems and other WLAN segments**
 - ✓ **uniform utilization of channels**

DFS implementation

We propose a DFS algorithm, which periodically discovers the best operating channel according to RSSI measures of background scanning. The associated and the contending stations send measurements to the AP. Based on its own measurement as well as the reports from the associated and contending stations, the AP makes the channel switching decision. Through this process the AP performs better area discovery and upgrades the link quality.

DFS experiment



Versions	Operating channel	Throughput (Mbps)	Improvement	
Average throughput	-	22.1	0%	-
MAdWifi	7	16.6	-	0%
DFS algorithm	1	28.2	30%	69%

DFS SCHEME FOR POWER EFFICIENCY

Contribution

- ▣ **In this work we focus on mitigating energy consumption on infrastructure IEEE 802.11 WiFi Networks.**
- ▣ **We propose:**
 - dynamic channel allocation techniques
 - energy conserving criteria
- ▣ **With these techniques we aim:**
 - to lead the WLAN system to a channel selection that lowers MAC layer frame retransmissions.

Greener DFS in WiFi Networks

- ▣ **Dynamic Frequency Selection was introduced in 802.11h amendment:**
 - for avoidance of co-channel operation with radars
 - uniform utilization of available channels.
- ▣ **With our work we give a new perspective in this field, by introducing new channel allocation techniques for power efficiency in WiFi networks.**

Green DFS Model

- ✓ We use the metric of *Contention Free Time* detected by the AP and the STAs.
- ✓ The decision of the channel that will be used is made by the AP, that gathers all the measurements.
- ✓ This decision is jointly based on the *Contention Free Time* and the network traffic of our WLAN.

Implementation

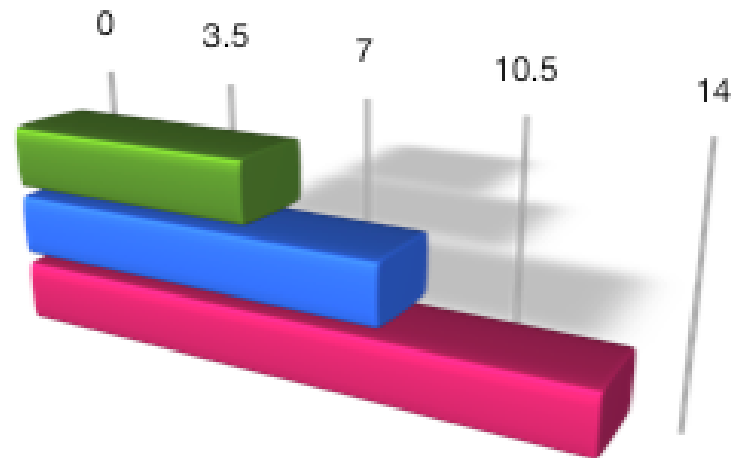
- ▣ **The evaluation of the performance of our modified MadWifi driver was executed on the NITOS wireless testbed.**
- ▣ **The energy efficiency of our DFS scheme is justified compared to the Static Weighted and the AP based model by means of retransmission percentage of the total UDP traffic generated frames.**

Experimentation Evaluation

▣ Frame Retransmission Percentage:

Our Scheme	Static Weighted	AP Based
5.9	8.9	13.2

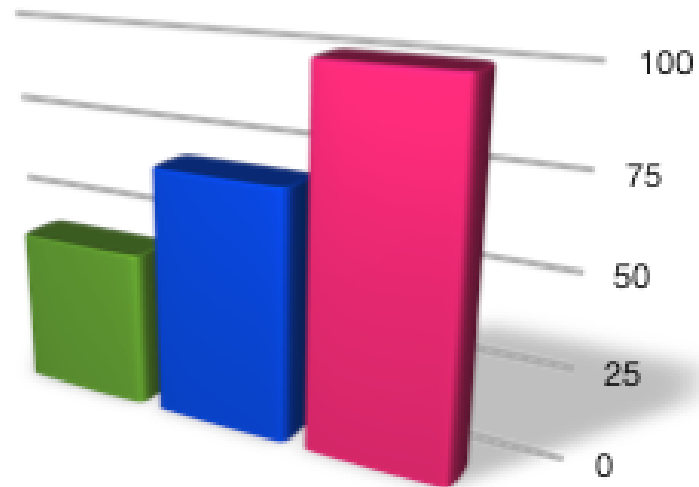
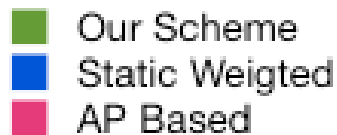
- Our Scheme
- Static Weighted
- AP Based



Experimentation Evaluation

▣ Power Consumption Percentage:

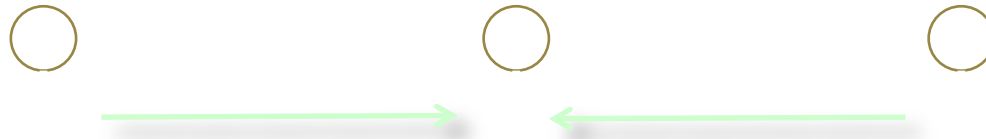
Our Scheme	Static Weighted	AP Based
43	68.4	100



RATE ADAPTATION -PACKET LOSS DIFFERENTIATION

NOISE & COLLISION

Collision



Noise



RATE ADAPTATION

Predict Channel Quality and choose the “right” rate/coding

Prediction based on Acknowledgments (ACKs):

Successful Transmission => increase rate

Failed Transmission => decrease rate

Why?

Noise is the sole reason for transmission failure, because:

802.11 -> 1997 - 1999

CSMA/CA -> collision avoidance

CSMA/CA

Carrier Sensing Multiple Access/Collision Avoidance

Avoidance: users perceive the medium differently

Backoff Timer: [0, CW]

“Packet Fair”

Problems (details in the end):

too many nodes

hidden terminal

capture effect

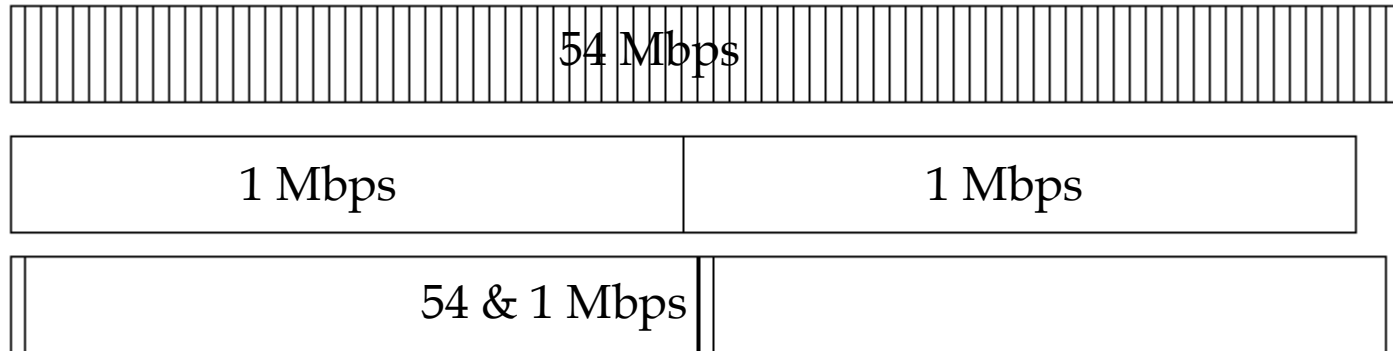
WRONG PREDICTION, SO WHAT?

Lower encoding = lower transmission rate =>

Lower throughput for the node itself

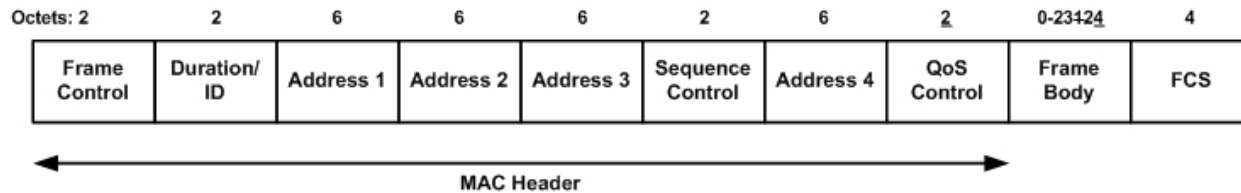
Lower throughput for the SYSTEM

Price of "Fairness" - 802.11 Anomaly



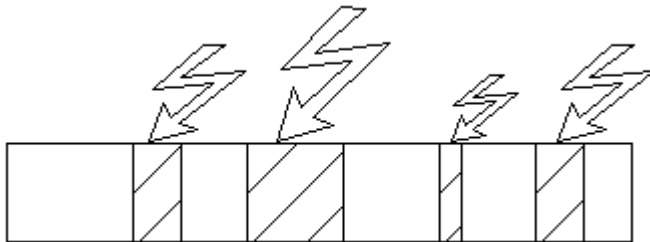
HOW TO DIFFERENTIATE?

Packet Frame



FCS: Frame Check Sequence

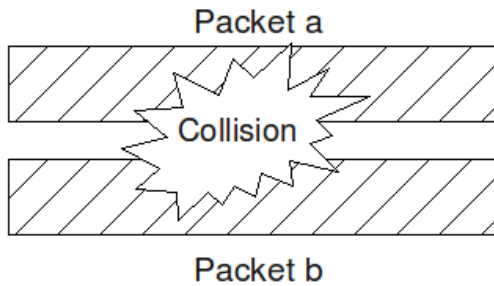
Packet destroyed by Noise:



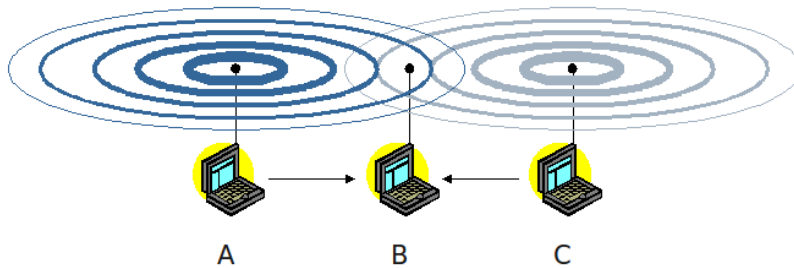
PACKET DESTROYED BY COLLISION

Collision scenarios:

Visible Nodes

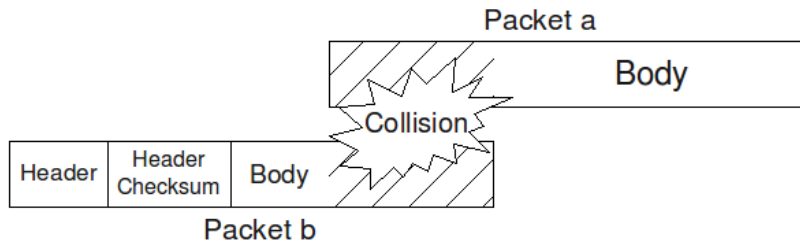
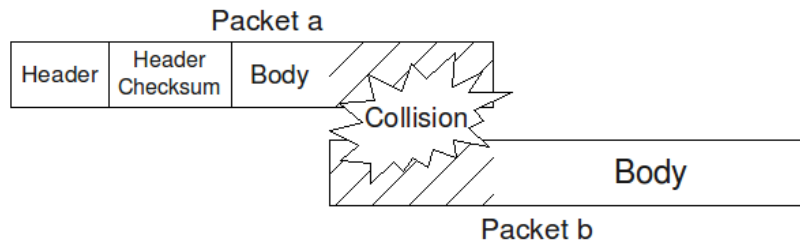


Hidden Terminal



PACKET DESTROYED BY COLLISION

Collision scenarios in Hidden Terminal:



THE IDEA

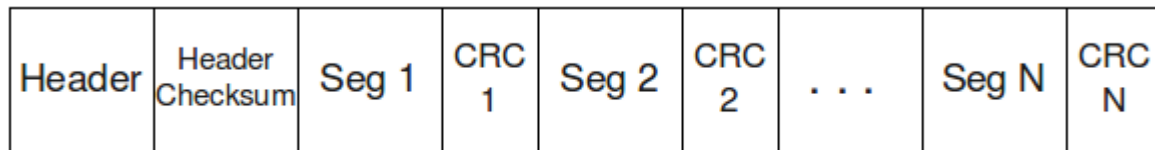
A more sophisticated checksum function

Smaller Segments (1500bytes/numOfSegs) =>

smaller CRC (16bits instead of 32) =>

small overhead in data transmitted and in CPU usage


Why distributed?



USE THE EXTRA INFO

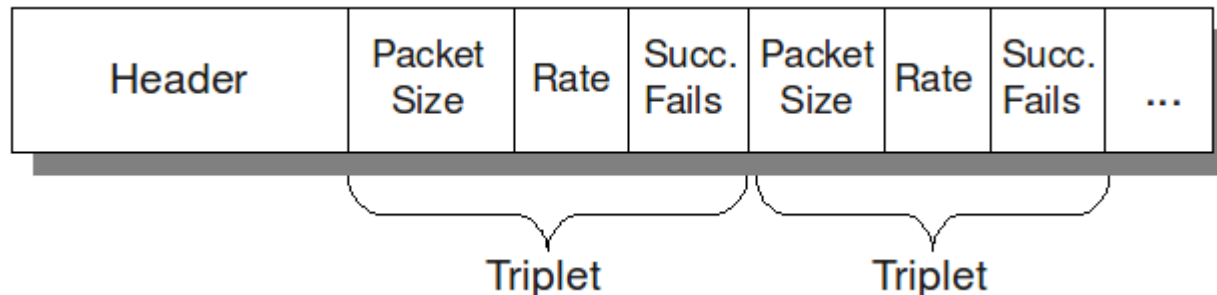
The sender knows the cause of failure but it must inform the sender *in time*

Successful Transmission => ACK =>  rate

“Noisy” Transmission => NAK =>  rate

Collision => - => -

NAK per node and when needed



WHY NAK ON NOISE?

Collision => half of the headers are destroyed

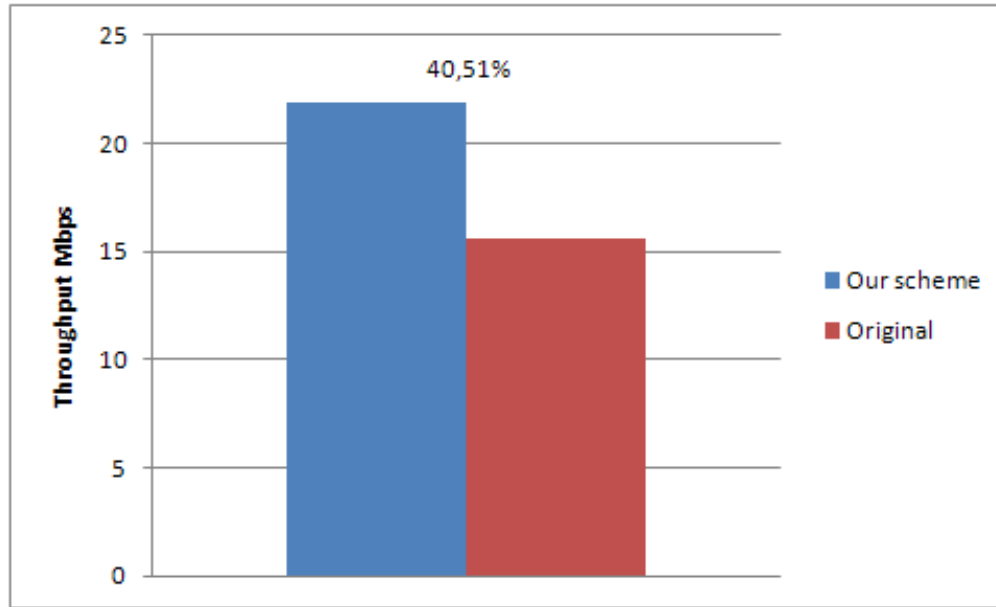
Receiver keeps per node statistics =>

header is needed =>

header small size (24bytes) =>

small possibility to be destroyed by noise (header XOR)

EXPERIMENTAL RESULTS



Experiments run in NITOS

A network with 6 nodes randomly distributed

USER ASSOCIATION

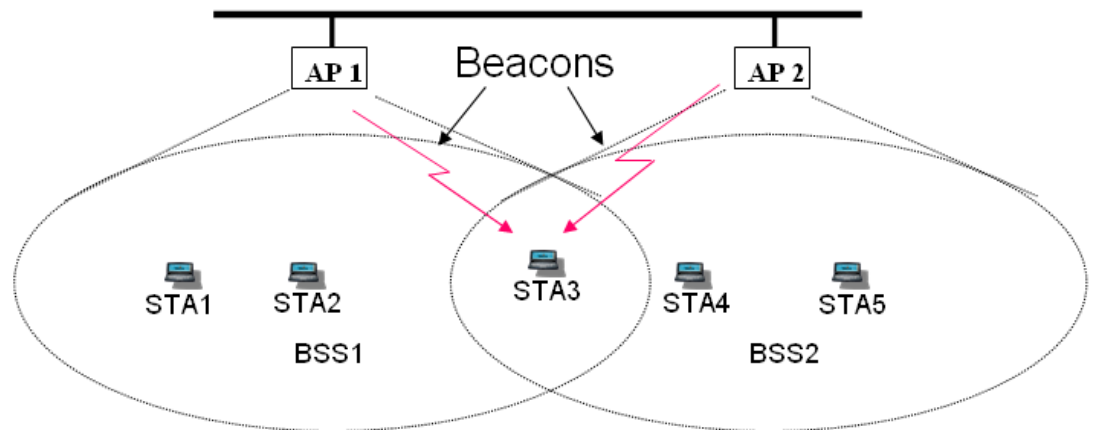
Association in 802.11 WLANS (1/2)

- In IEEE 802.11 WLANs, each station (STA) has to first associate with an access point (AP), before it can start transmitting data to other nodes in the network.
- During the Decision Phase the STA determines the AP that is the most appropriate to associate with
- IEEE 802.11 standard

defines

**RSSI - based
Association**

Policy



- A STA simply selects the AP from which it has received the strongest signal during the scanning process.

Association in 802.11 WLANS

(2/2)

- **Main problems in the standard mechanism:**
 - RSSI is not an appropriate decision factor for user association (high RSSI values cannot univocally indicate high throughput)
 - RSSI is an indicator for the **Downlink** but not for the **Uplink** channel conditions
 - End user performance relies on several factors, except from channel conditions such as:
 - **ChannelContention:** contending nodes and their individual PHY rates.
 - **APLoad:** associated STAs and their individual PHY rates.
 - **Interference** on the channel an AP offers.
- **A comprehensive metric is needed, that jointly considers all the above parameters.**

Throughput in CSMA/CA networks Contention Effect

$$T_n = \frac{1}{\frac{1}{R_n} + \sum_{k=1}^{|A_n|} \frac{1}{R_k}}$$

When there are multiple transmissions with different PHY rates used (R_k) in the 1-hop neighborhood of transmitter node n (A_n), each node enjoys an equally shared value of throughput T_n .

The Proposed Metrics (1/2)

We begin from the basic equation that calculates the affordable throughput in 802.11 networks that consist of several nodes contending for channel use and extend it by integrating the effects induced by:

1. Associated Stations:

The bandwidth the each AP can offer on downlink is shared among its associated STAs.

2. Interfering nodes:

In a simple downlink scenario, interference is caused of active hidden terminals at the STA's side. We capture this effect by modeling the corresponding decrease in performance.

The Proposed Metrics (2/2)

3. Traffic estimation:

We contribute by incorporating in our metrics the characterization of each transmitter node by its traffic activity.

Our implementation provides though proper messaging the exchange of traffic information among neighboring nodes.

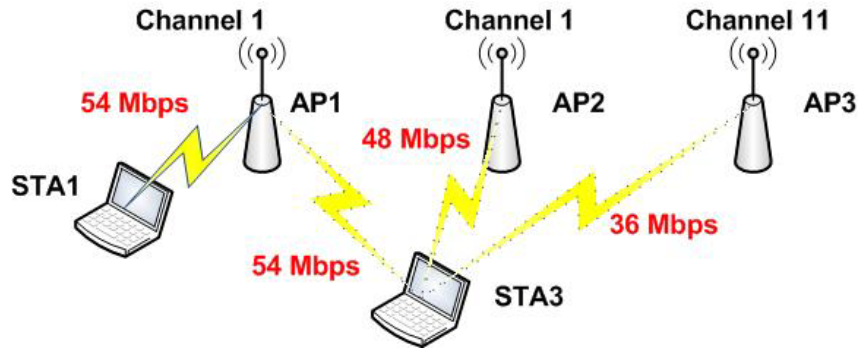
4. Separate metrics for Downlink and Uplink Channels

The contention and interference conditions vary for each transmitter of the of the network and as a result they are different for the uplink and downlink channels.

$$T_{ij}^{total} = u_r \cdot T_{ij}^{up} + d_r \cdot T_{ij}^{down}$$

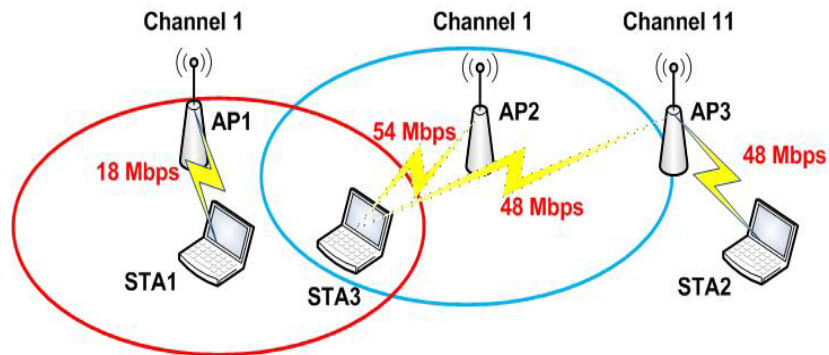
Having collected information for each available AP_j , STA_i calculates the combined metric for each possible association according to its application requirements by setting u_r , d_r and finally decides to associate with the AP_j, that provides the maximum calculated metric.

Experiments (1/3)



- **RSSI:** STA3 associates with AP1 (15,5 Mbps)
- Considering only the operation of associated STAs, STA3 associates with AP2, avoiding AP1 due to the operation of associated STA1 (18 Mbps).

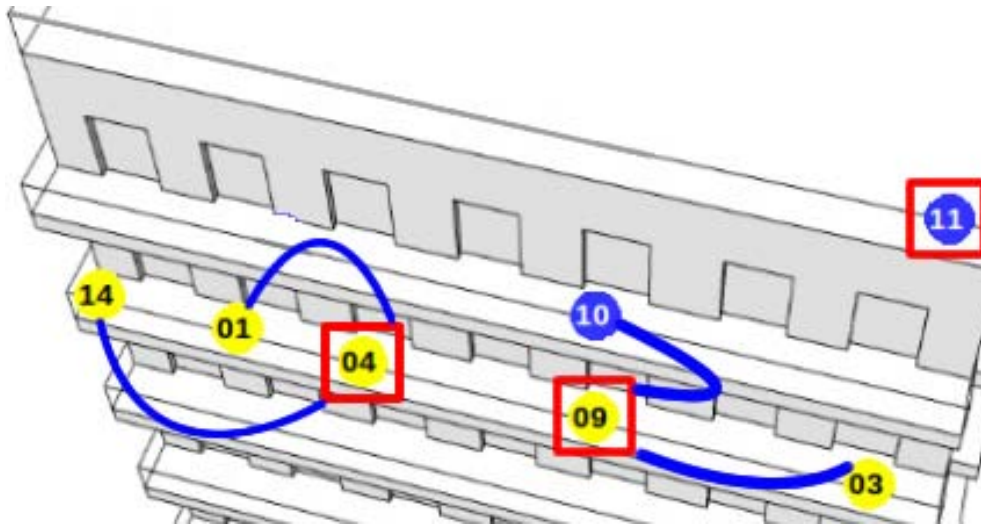
- **PROPOSED:** STA3 associates with AP3, avoiding AP2 due to the operation of the contending STA1 (20,5 Mbps).



- **RSSI:** STA3 associates with AP2 (4,5 Mbps)
- But STA1 appears to be hidden to the transmissions of AP2 destined to STA3, leading to degraded PDR at the receiver STA3.

- **PROPOSED:** considering the operation of active interfering nodes, STA3 associates with AP3, leading performance to 24 Mbps (6x).

Experiments (2/3)



•In this experiment nodes 04, 09 and 11 operate as APs and the rest nodes operate as STAs.

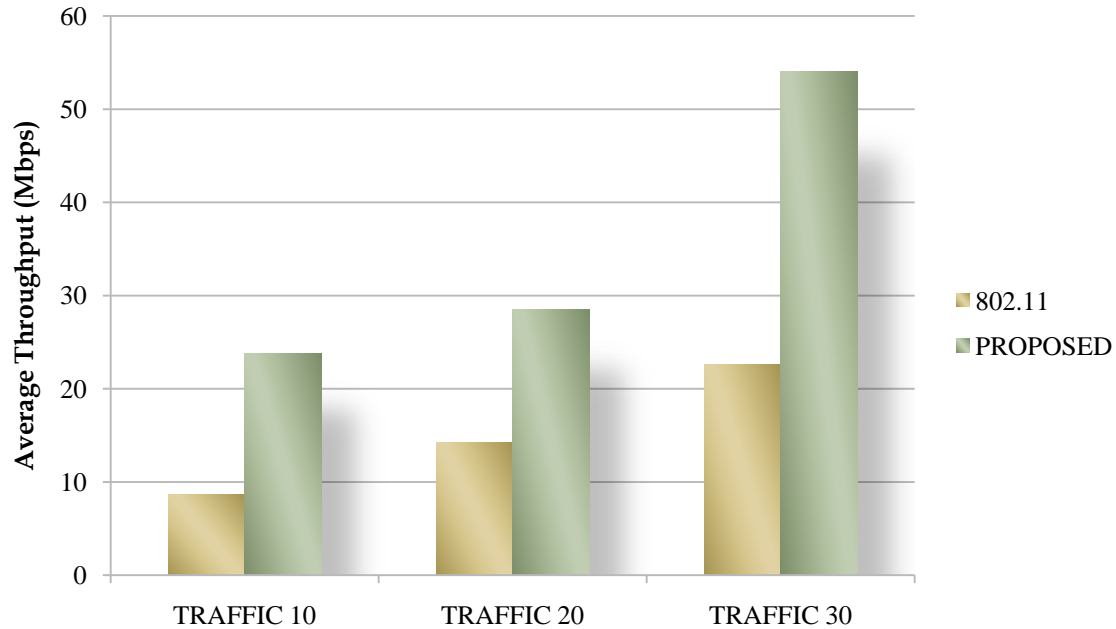
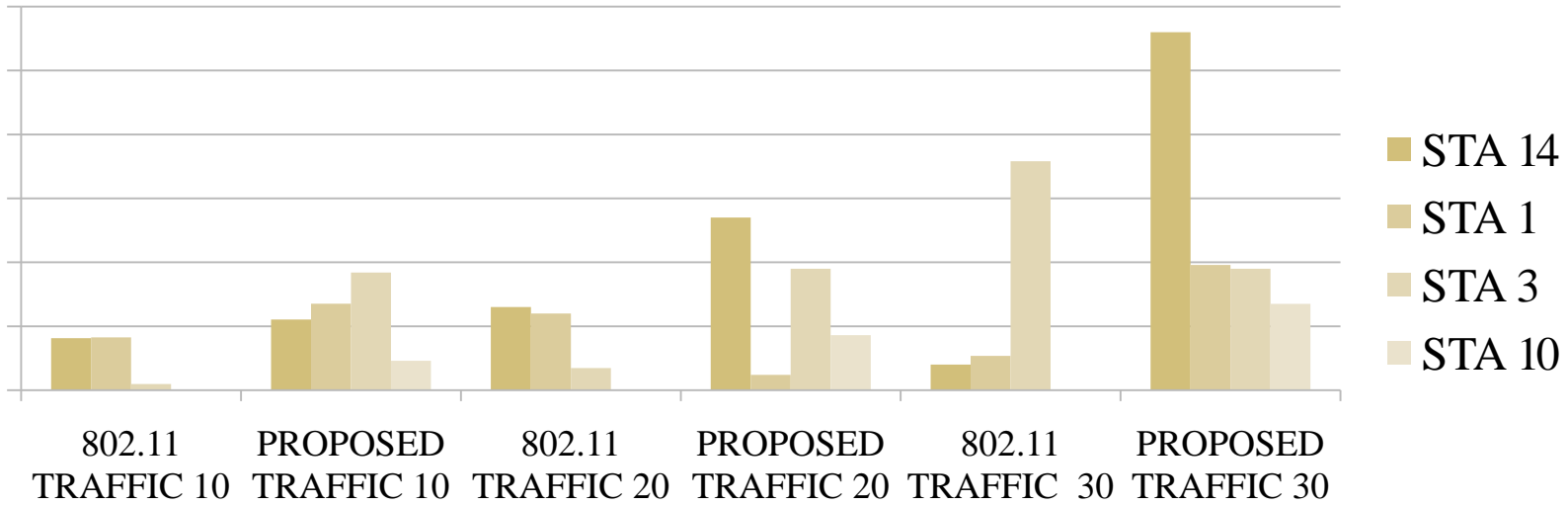
•In the first figure blue lines represent the resulting associations, by following the standard approach.



•In the second figure blue lines represent the associations that existed for greater duration, while green lines represent associations that existed for smaller duration, by following our approach.

Experiments (3/3)

Average
Throughput (Mbps)

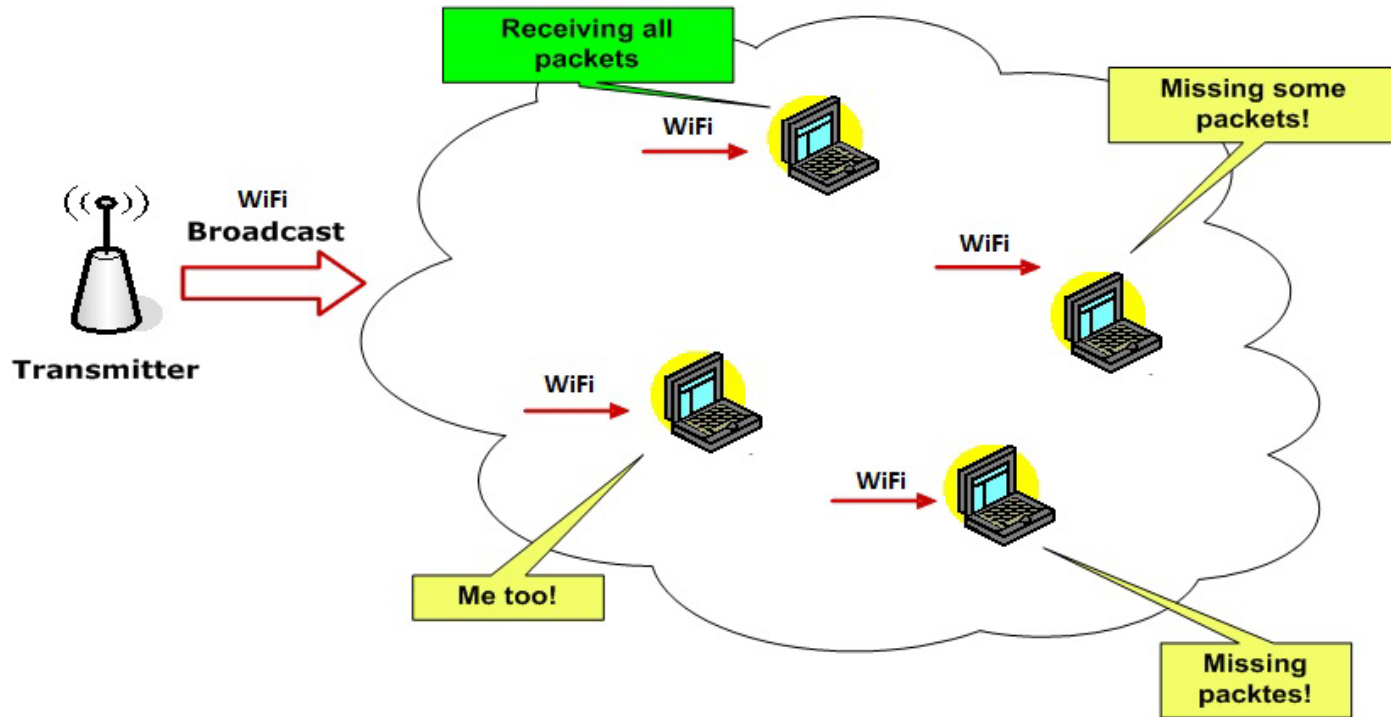


COOPERATIVE NETWORKS

Cooperative scheme

- **Demand for multimedia services is rapidly growing e.g. MobileTV**
- **Need for multicast/broadcast capabilities for the mass distribution of multimedia services e.g. 3GPP MBMS**
- **Difficult to guarantee the reception reliability**
 - **Wireless Medium is highly error prone and unpredictable**
 - **Each receiver experiences heterogeneous channel conditions**
- **No reverse communication channel for the receiver to request retransmission of lost data frame**
- **Key challenge is to provide good quality multicast/broadcast service for multiple receivers**
- **Solution ?**

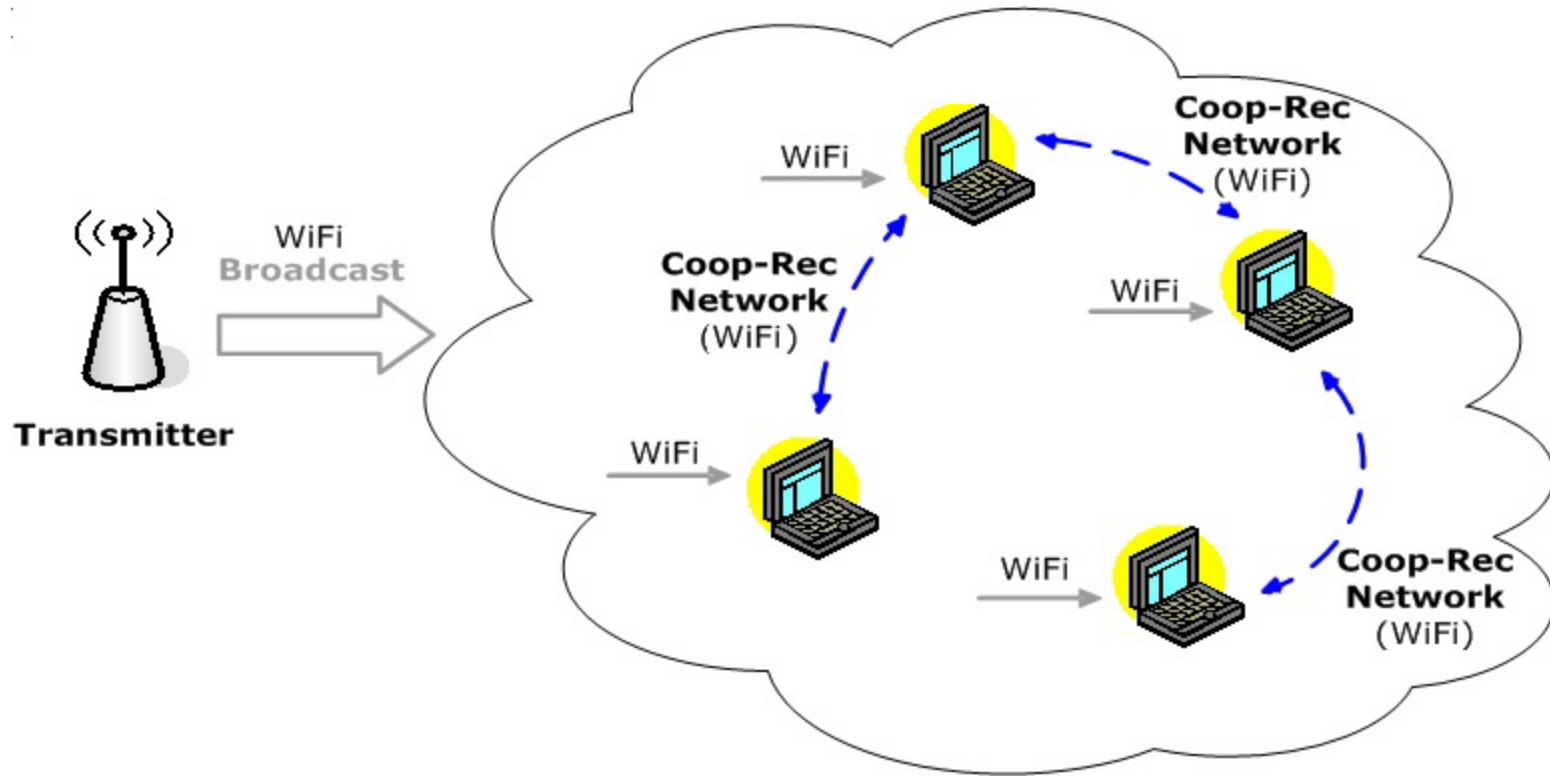
Cooperative scheme



Communication from the first WiFi interface

Devices may experience data losses due to heterogeneous wireless channel conditions

Cooperative scheme



✓ Communication from the second WiFi interface

✓ Devices run the Cooperative Recovery protocol to dynamically form an cooperative recovery (ad hoc) network and recover the lost multicast data packets from other peers

OTHER ACTIVITIES

VIDEO OVER WIRELESS

Video over wireless scheme Framework

15 of the wireless nodes are equipped with Logitech C120 USB web-cameras.



- NITOS aims to provide a Video Framework to support:
 - Monitoring and capturing real time video streams
 - QoS for video streaming in wireless networks
 - Cooperation protocol for reliable multicast video streams
 - Easily configuration of coding/decoding transmitted frames

Video over wireless scheme

User Cases

- ▣ One-to-one wireless video transmission in a multi-hop environment (one video transmitter, remote video player)
- ▣ One-to-many wireless video transmission in a multi-hop environment (one video broadcasting transmitter, multiple receivers-players)
- ▣ Many-to-one wireless video transmission... (multiple video transmitters, one receiver-server)

Video over wireless scheme

User Cases

- ▣ Develop cooperative scenarios (enable relays)
- ▣ Change the access mechanism (and generate scenarios that are not based on contention, e.g. synchronized transmission among neighbors in the broadcasting - cooperative mode)
- ▣ Filter the packets in any of the relays (forward part of the video, based on the video coding)
- ▣ Develop application layer FEC

POWER METER INFRASTRUCTURE

Power Meter Scheme

- ▣ **We are in the process of developing a power metering scheme that will give us energy measurements of a building**
- ▣ **We develop a software framework that will let us store the power consumption data in a local database and depict in graphs any consumption period selected by the user**

How we take our measurements (1/4)

- ▣ In the facilities of CERTH in Volos we have installed a CurrentCost Power Meter, which is used to measure total energy consumption.



How we take our measurements (2/4)

- ▣ Every 6 seconds power data are sent wirelessly to a receiver which is also equipped with an LCD display.



How we take our measurements (3/4)

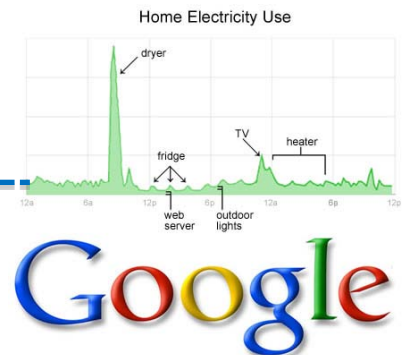
- ▣ The receiver is connected via USB to a PC.
- ▣ The PC sends, through the internet, the power meter data to the cloud service of Google Powermeter.



USB

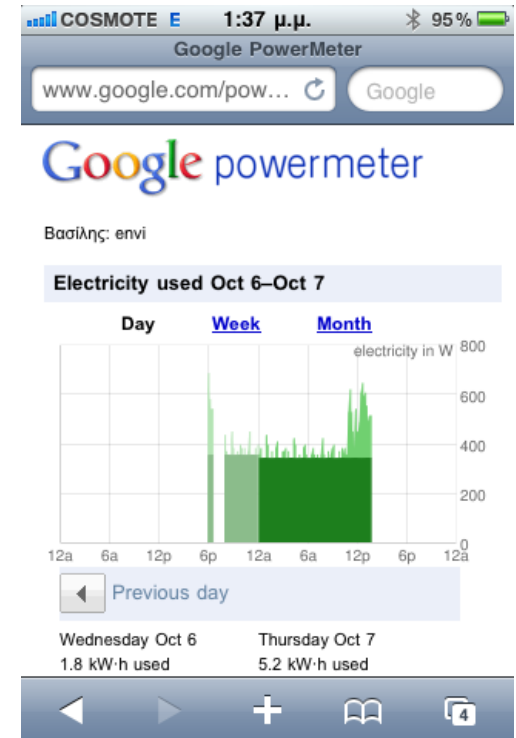
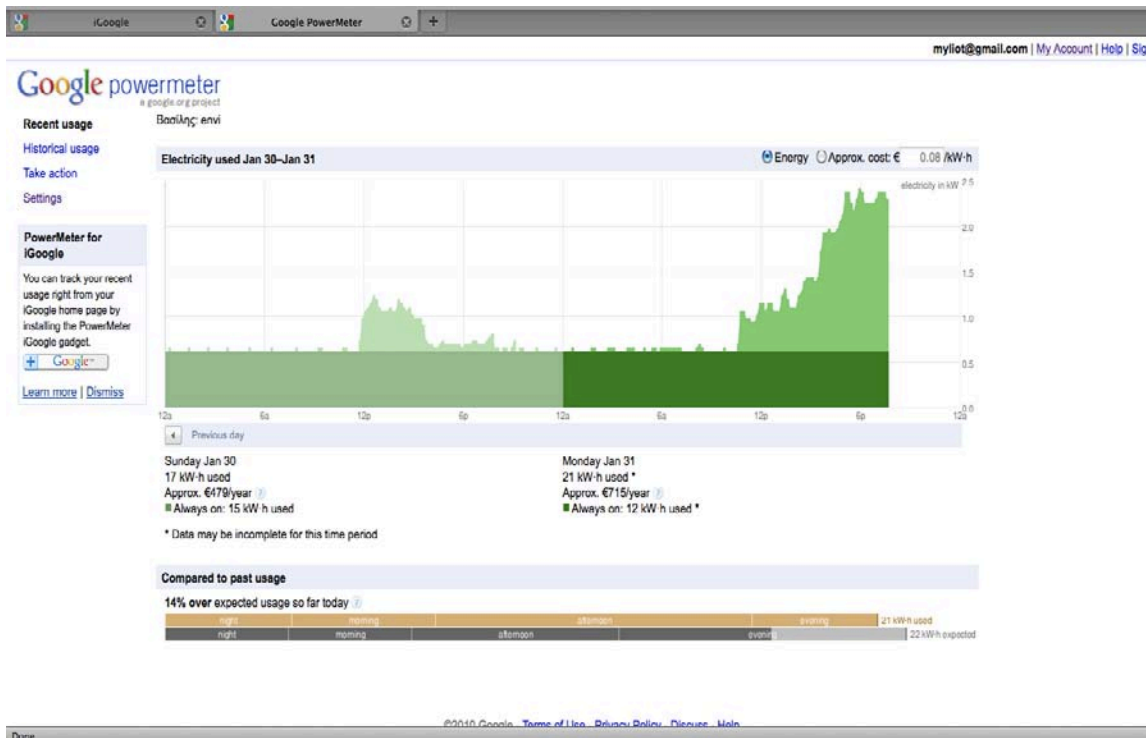


Internet



How we take our measurements (4 / 4)

- ▣ The data collected in the cloud can then be seen on a web browser or a smartphone.

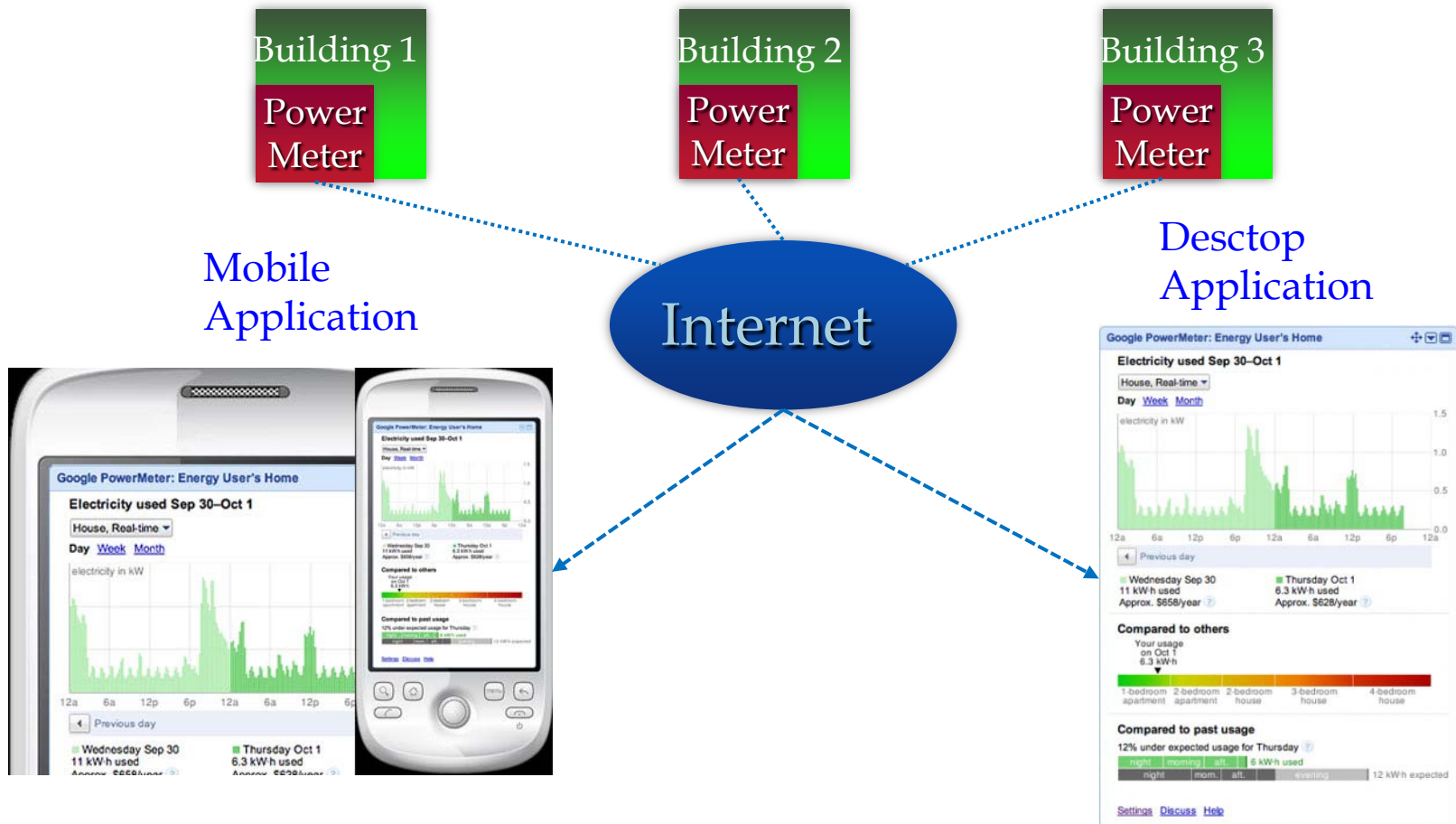


Power Meter Scheme

- ▣ **We will deploy the power meters throughout the buildings of the University of Thessaly campus**
- ▣ **In this way we will be able to track the energy consumption of the whole university**
- ▣ **In points of measurements that have no internet access we will connect our meters with wireless nodes in order to transmit the data to the aggregate database**

GreenUTH

Integration of power consumption data



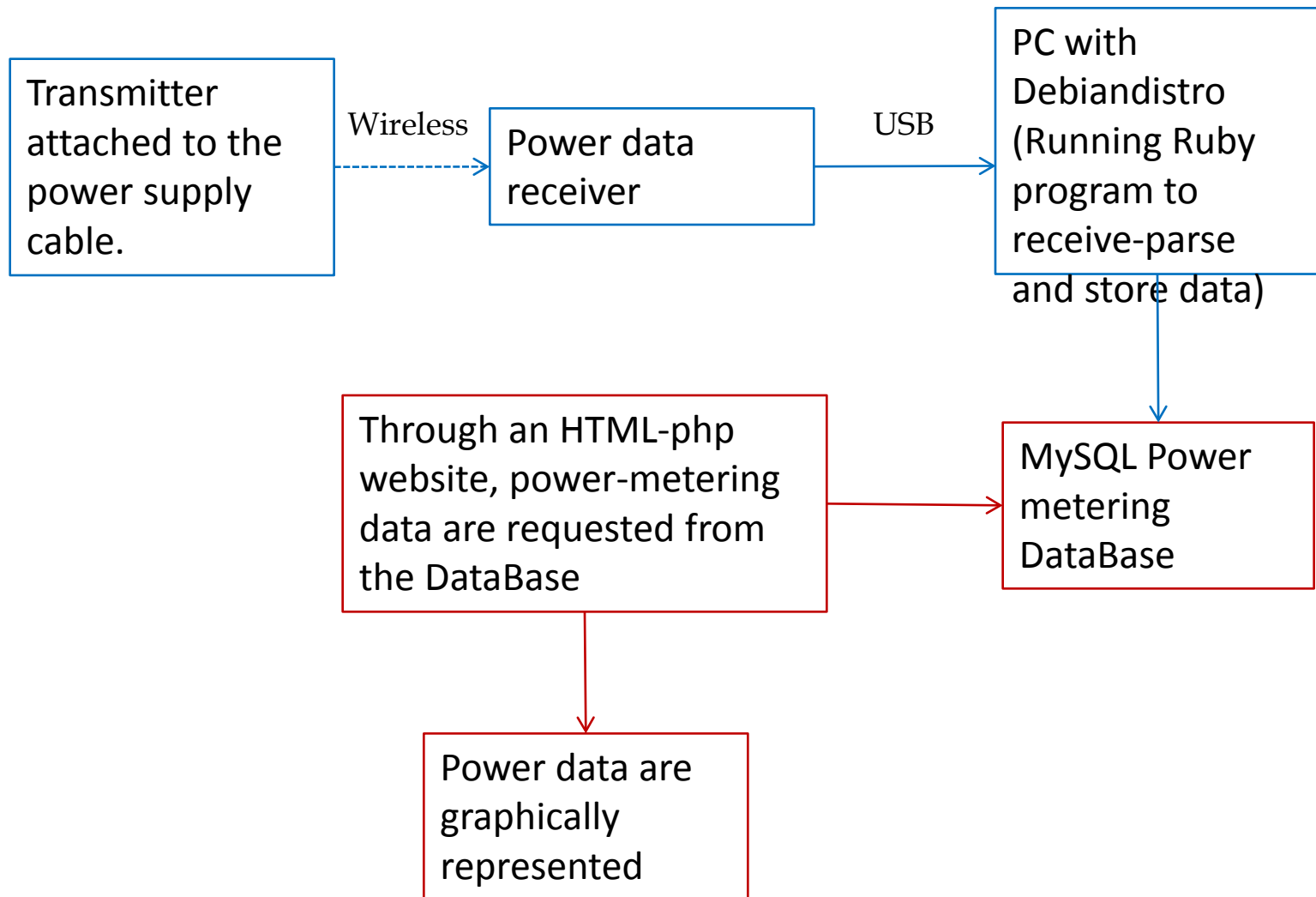
Power Meter Open Source

- ▣ **The whole framework is based on Open Source software**
- ▣ **We use Ruby language to receive and parse the consumption data from the power metering devices**
- ▣ **MySQL is used in order to store the data locally without the need of web power metering storage services**

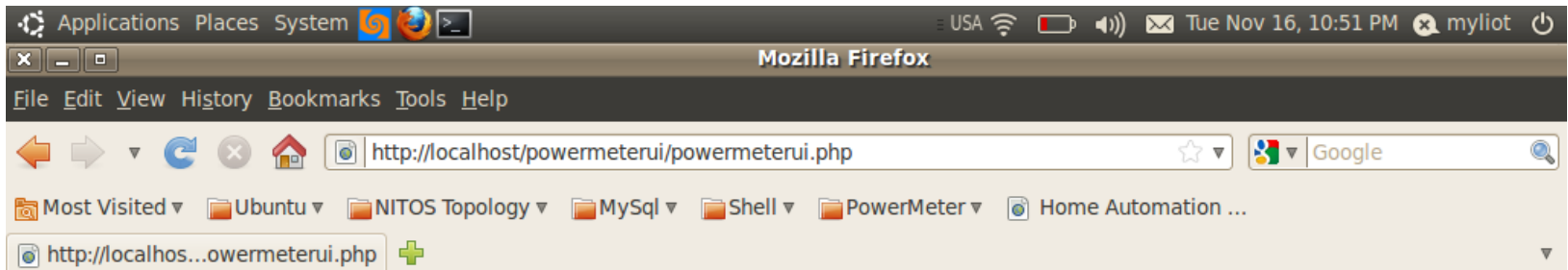
How we collect our data (1/4)

- ▣ **We have developed a Ruby program in order to read an .xml file that is received by the PC every 6 secs from the USB port.**
- ▣ **We use a secure MySQL database to store our power metering data.**
- ▣ **An HTML - php website using pCharts is developed to graphically represent energy consumption data.**

How we collect our data (2/4)



How we collect our data (3/4)



Choose start date & time

[Pick date](#)

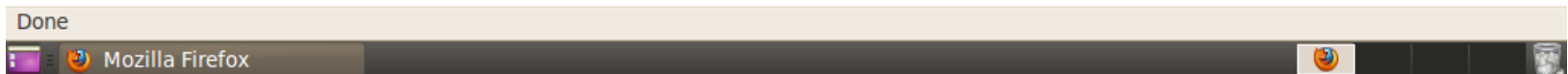
[Pick time:](#) hh min

Choose end date & time

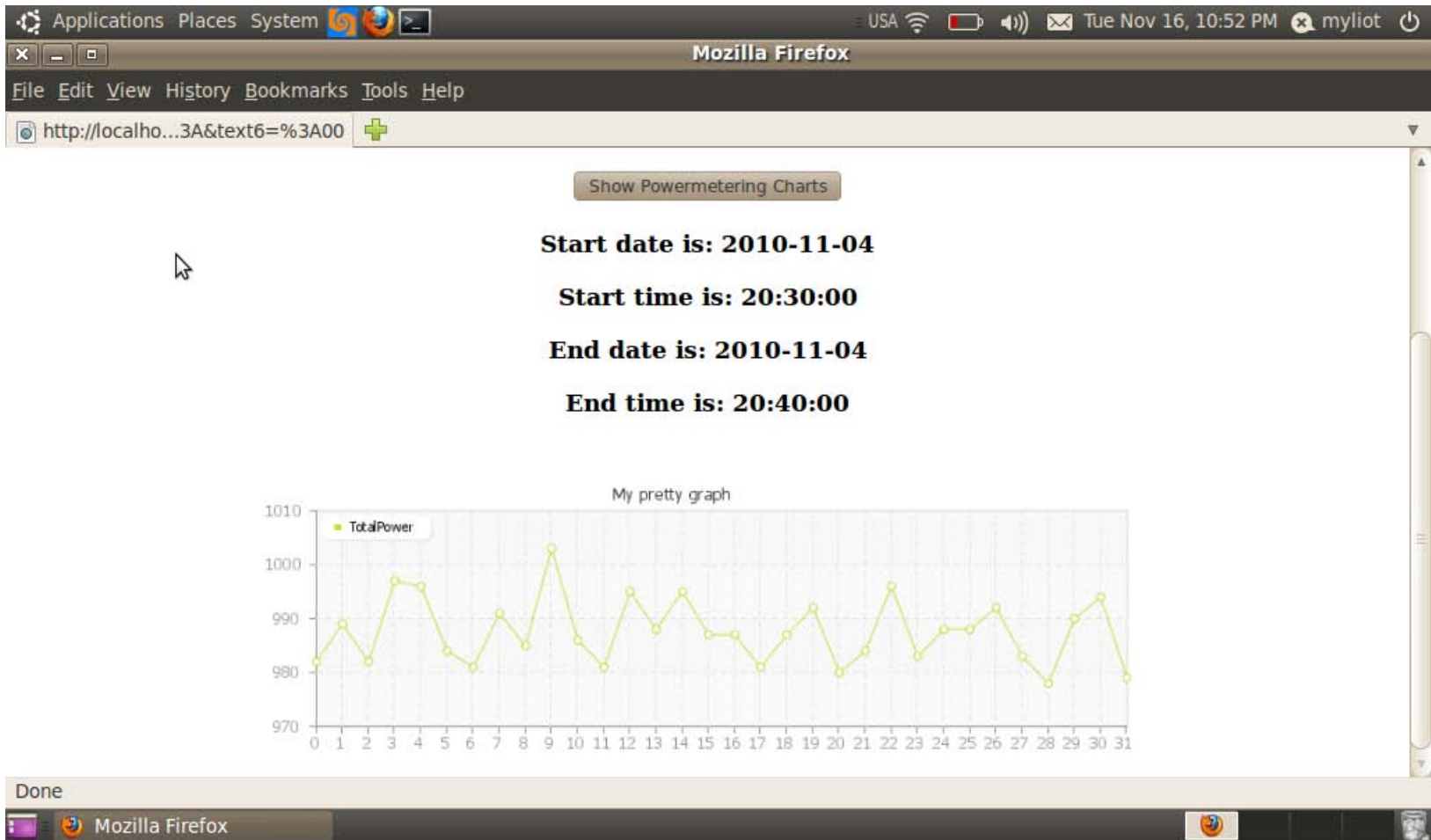
[Pick date](#)

[Pick time:](#) hh min

Show Powermetering Charts



How we collect our data (4/4)



Power Meter Interventions

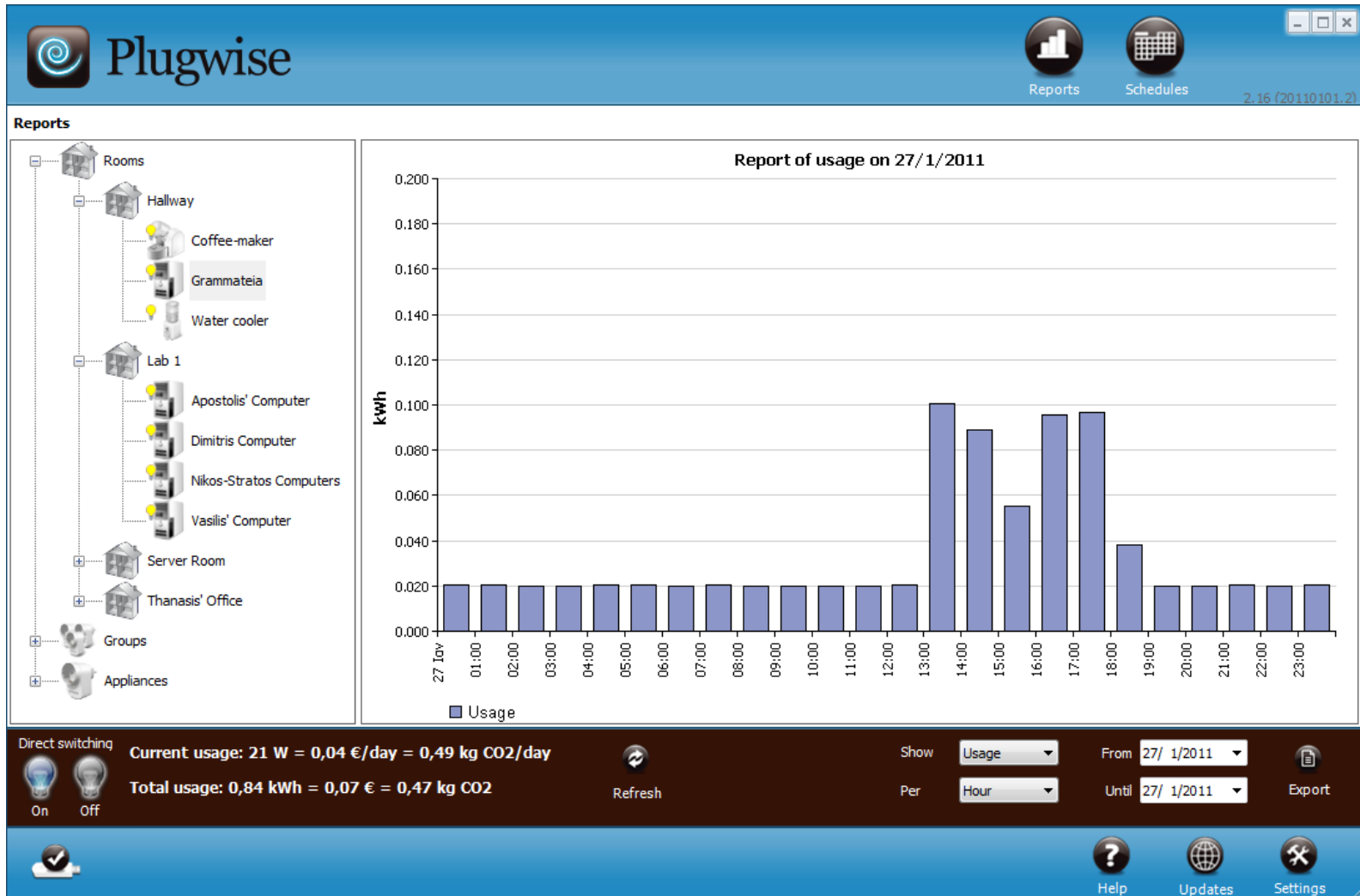
- ▣ **We are also deploying ZigBee plugs and switches in order to be able to measure separately the consumption of electric appliances in a building**
- ▣ **At the same time we are building a framework in order to be able to remotely switch ON and OFF any electric appliance**
- ▣ **Presence sensors will be in charge of automatically switching off any electric appliance that could be forgotten ON.**

ZigBee plugs (1/2)

- ▣ These plugs are inserted between wall plugs and appliances.
- ▣ They can be scheduled to turn off during a chosen period of time or if the consumption is below a pre-specified threshold to save standby energy consumption



ZigBee plugs (2/2)

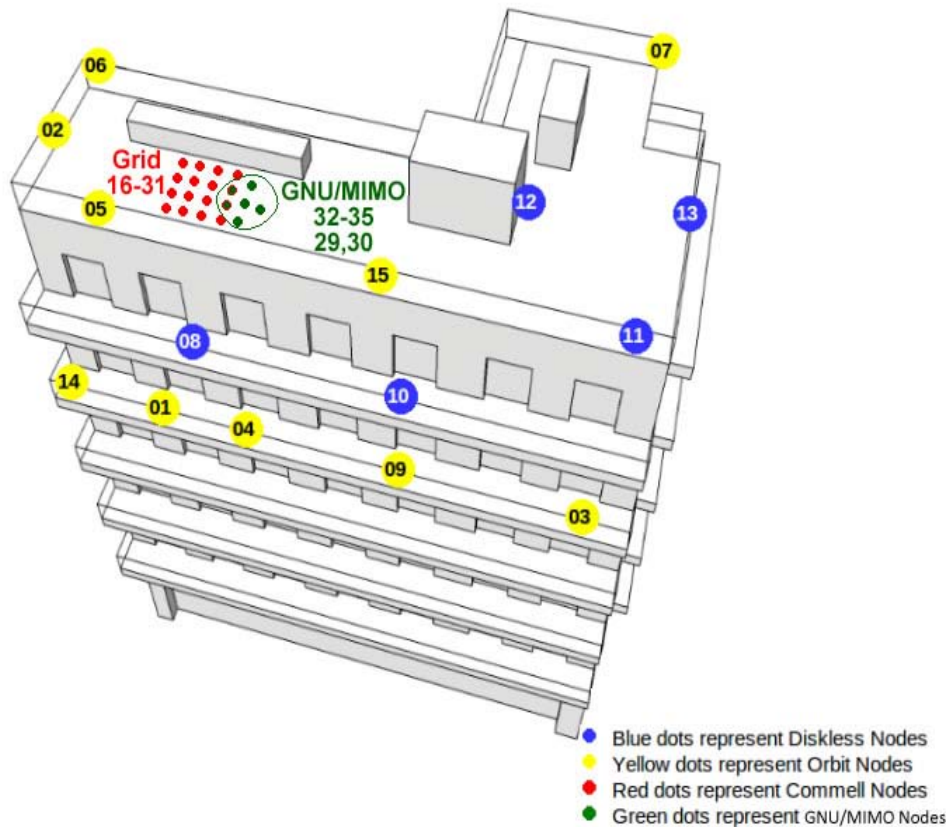


Power Meter Scheme

- ▣ **Being able to measure the whole and separate consumption in a building, we will explore its power consumption profile**
- ▣ **In this way we will extend our means of intervention for greener buildings by revealing waste of energy and the cause that provokes this waste**

SENSORS FRAMEWORK

NITlab Sensors Framework (1/6)



Each ORBIT node of NITOS is equipped with a Temperature & Humidity USB Sensor THUM (Temperature and HUmidity Monitor)

NITlab Sensors Framework (2/6)

NITOS provides its users with the ability to run experiments using the temperature and humidity sensors (online) or even access **real sensor records** by using the **NITOS Sensors Toolkit** (offline).

The filesystem provided for sensor experiments contains scripts that help the user easily setup an experiment, just by providing the reserved sensor IDs, the measurement period and the total number of measurements that will be recorded.

```
SENSOR: voyage-sensors/measurements/h/h015.txt
AVG: 56.085433
VAR: 0.023897378618

Number of Sensors: 10
Measurements per Sensor: 5

Expected Mean per Sensor:
80.3089316 81.6573486 81.3460548 79.2107048 80.4107766 50.0935926 38.894159 61.0951702 50.8923228 56.085433

Expected Mean per Measurement Round:
70.5319309 70.9961124 70.8593423 71.3576071 71.2522543
```

Moreover, each user can use special scripts to transfer the measurements taken during his reservation period to the **NITOS measurements database for posterior processing**.

NITlab Sensors Framework (3/6)



Home Testbed Scheduler Projects News Contact us

Welcome to the NITOS Sensors Toolkit

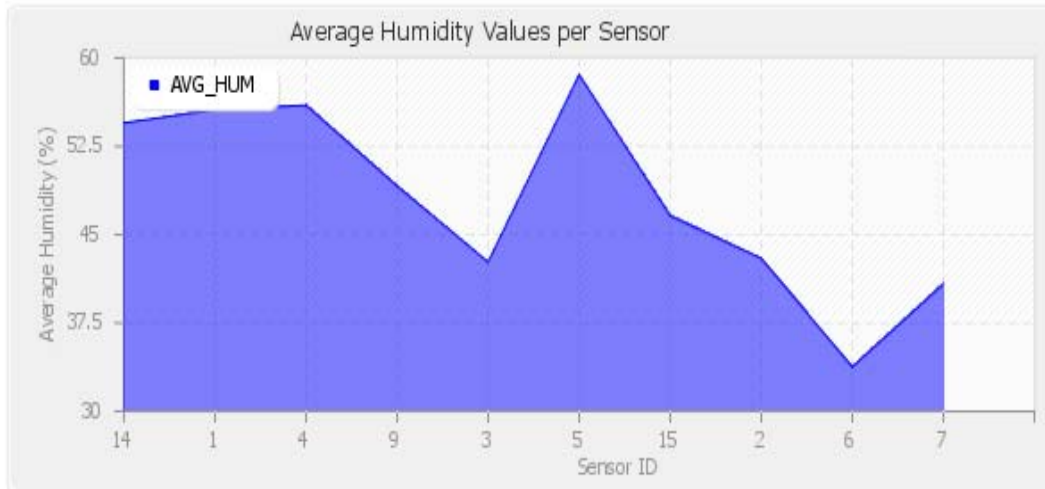
Please provide any combination of the following fields to see a list of the available Measurement Sets

Username: Period: Measurements:

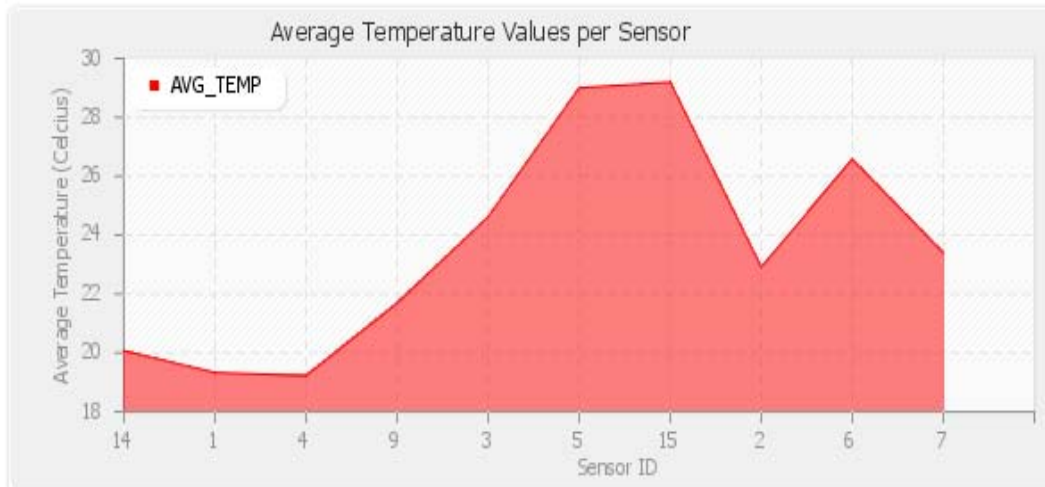
SET_ID	USERNAME	START TIME	PERIOD (secs)	# OF SENSORS	# OF MEASUREMENTS
150	efkerani	2010-12-08 19:17:02	180	10	20
149	efkerani	2010-12-08 14:57:48	60	10	20
75	efkerani	2010-12-03 19:42:45	300	10	14
74	efkerani	2010-12-03 19:42:31	5	5	5
73	efkerani	2010-12-03 19:29:53	5	4	5

Select a specific Set_ID to get a graphical representation of its measurements:

NITlab Sensors Framework (4/6)

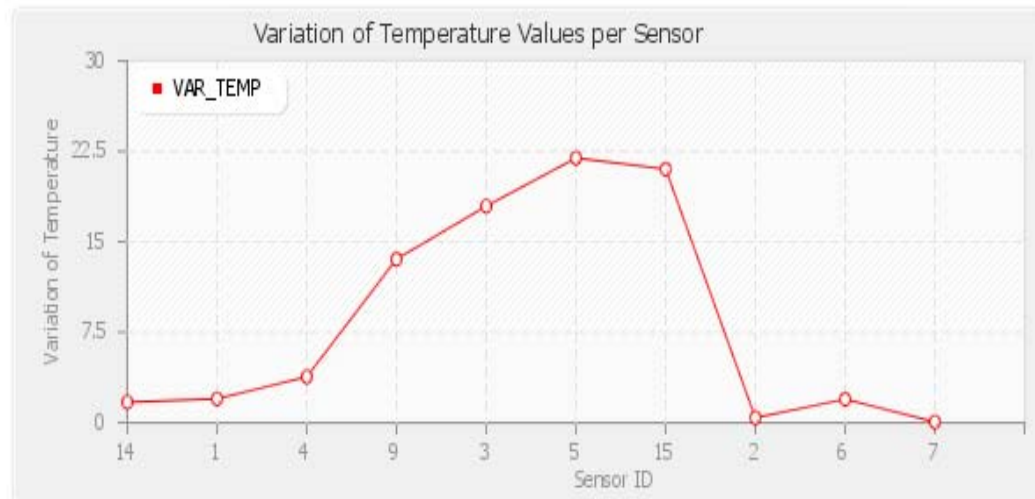
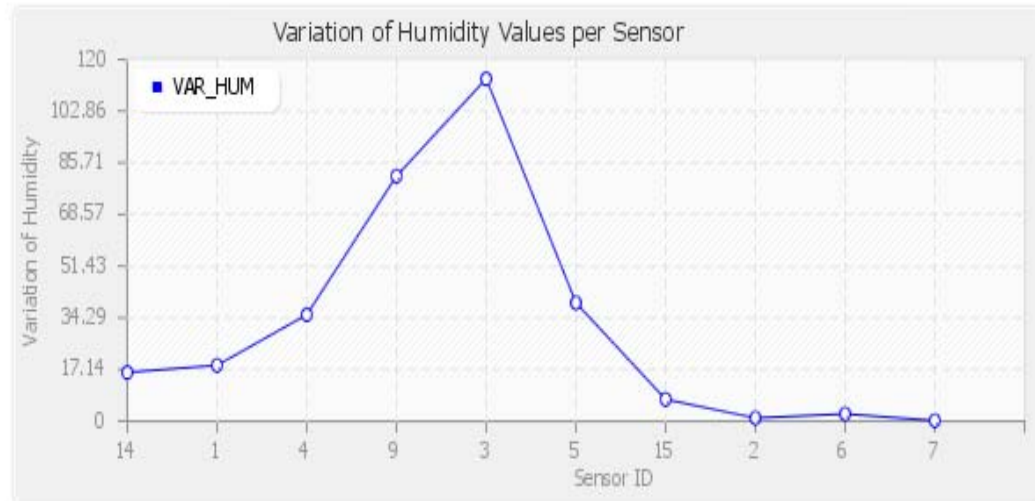


Using the **Sensors Toolkit** the user can get a **graphical representation** of each measurement set stored in NITOS database.



Various statistical measures are extracted from the corresponding records, such as **average humidity and temperature values** per sensor or per measurement round.

NITlab Sensors Framework (5/6)



NITlab Sensors Framework (6/6)

Correlation Matrix for Humidity Measurements

NODE_ID	14	1	4	9	3	5	15	2	6	7
14	1.000	0.632	0.894	-0.208	-0.397	0.125	-0.111	0.191	0.026	0.243
1	0.632	1.000	0.793	0.109	-0.322	0.319	0.174	0.160	0.144	0.154
4	0.894	0.793	1.000	-0.152	-0.450	0.361	0.013	0.329	0.308	0.158
9	-0.208	0.109	-0.152	1.000	0.510	0.247	0.651	-0.604	-0.430	-0.301
3	-0.397	-0.322	-0.450	0.510	1.000	0.009	0.390	-0.590	-0.347	-0.417
5	0.125	0.319	0.361	0.247	0.009	1.000	0.703	-0.268	0.429	-0.700
15	-0.111	0.174	0.013	0.651	0.390	0.703	1.000	-0.640	-0.034	-0.751
2	0.191	0.160	0.329	-0.604	-0.590	-0.268	-0.640	1.000	0.632	0.434
6	0.026	0.144	0.308	-0.430	-0.347	0.429	-0.034	0.632	1.000	-0.272
7	0.243	0.154	0.158	-0.301	-0.417	-0.700	-0.751	0.434	-0.272	1.000

Correlation Matrix for Temperature Measurements

NODE_ID	14	1	4	9	3	5	15	2	6	7
14	1.000	0.743	0.969	0.012	-0.200	0.649	0.284	0.326	0.534	0.220
1	0.743	1.000	0.822	0.232	-0.109	0.604	0.419	0.207	0.409	0.204
4	0.969	0.822	1.000	0.024	-0.251	0.664	0.285	0.399	0.589	0.345
9	0.012	0.232	0.024	1.000	0.685	0.505	0.764	-0.607	-0.202	-0.491
3	-0.200	-0.109	-0.251	0.685	1.000	0.322	0.638	-0.720	-0.313	-0.491
5	0.649	0.604	0.664	0.505	0.322	1.000	0.787	-0.097	0.473	-0.050
15	0.284	0.419	0.285	0.764	0.638	0.787	1.000	-0.469	0.099	-0.454
2	0.326	0.207	0.399	-0.607	-0.720	-0.097	-0.469	1.000	0.758	0.701
6	0.534	0.409	0.589	-0.202	-0.313	0.473	0.099	0.758	1.000	0.573
7	0.220	0.204	0.345	-0.491	-0.491	-0.050	-0.454	0.701	0.573	1.000

One important feature that the toolkit provides is the correlation matrix tool that provides an indication of how much the measurements of each sensor are correlated with each other.

Currently we are experimenting with the integration of the spatial correlation among the sensors in the:

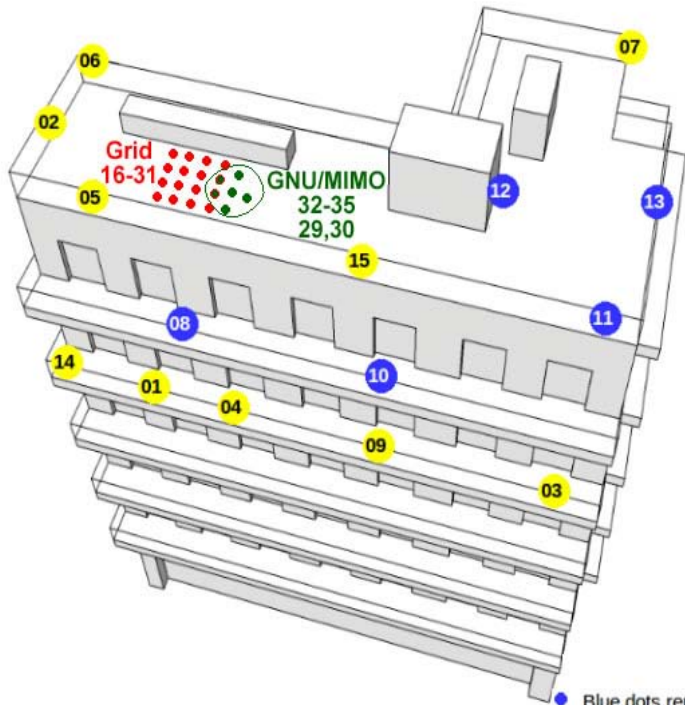
- Channel access mechanism of sensor networks.
- In-network Data Aggregation schemes

in order to reduce:

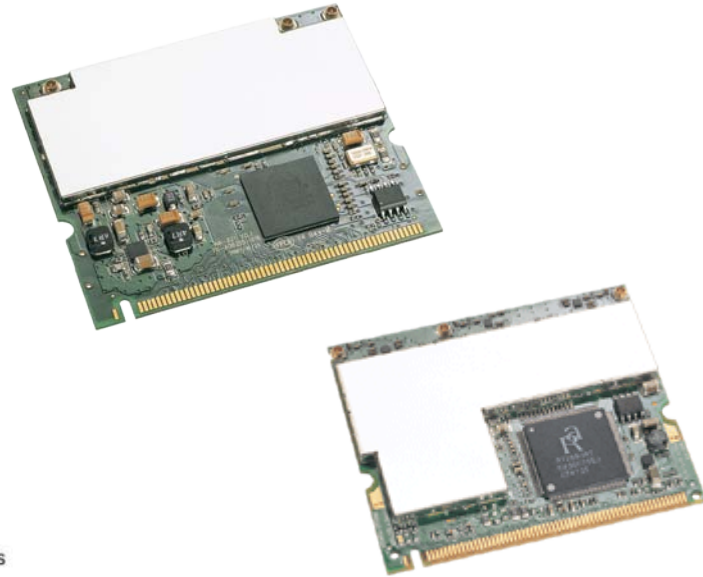
- levels of contention
- redundant data transmissions
- overall energy consumption

MIMO FRAMEWORK

MIMO Framework (1/2)



- Blue dots represent Diskless Nodes
- Yellow dots represent Orbit Nodes
- Red dots represent Commell Nodes
- Green dots represent GNU/MIMO Nodes



6 nodes of NITOS (green) are equipped with MIMO enabled wireless interfaces

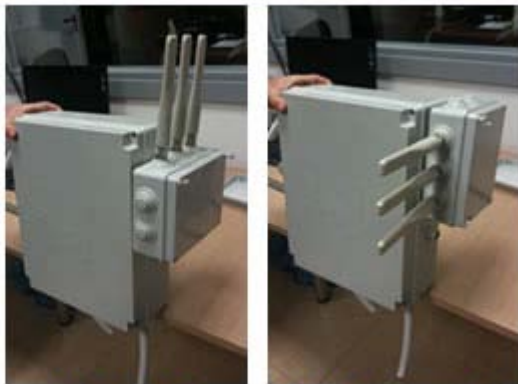
MIMO Framework (2/2)



Wireless cards:

- WMIA-199N WLAN 802.11 n Atheros 9160
- WMIR-200N Ralink RT2860+2850

Two types
of
Antennas
placement
for the
MIMO
nodes



Vertical

Horizontal

All MIMO-enabled nodes of NITOS are currently equipped with an extra box that has 3 Dual Band 2.4/5GHz, 7dbi Antennas mounted. The antenna box is also able to turn, so that the experimenter can create more specific topologies by exploiting the polarization of the antennas.

THANK YOU