The EXPRESS SDN Experiment in the OpenLab Large Scale Shared Experimental Facility

Serge Fdida

UPMC, Sorbonne University & CNRS

Special Thanks to the contributors:

Stefano Salsano, Giuseppe Siracusano, CNIT, Italy

Thanasis Korakis, Harris Niavis, UTH Greece

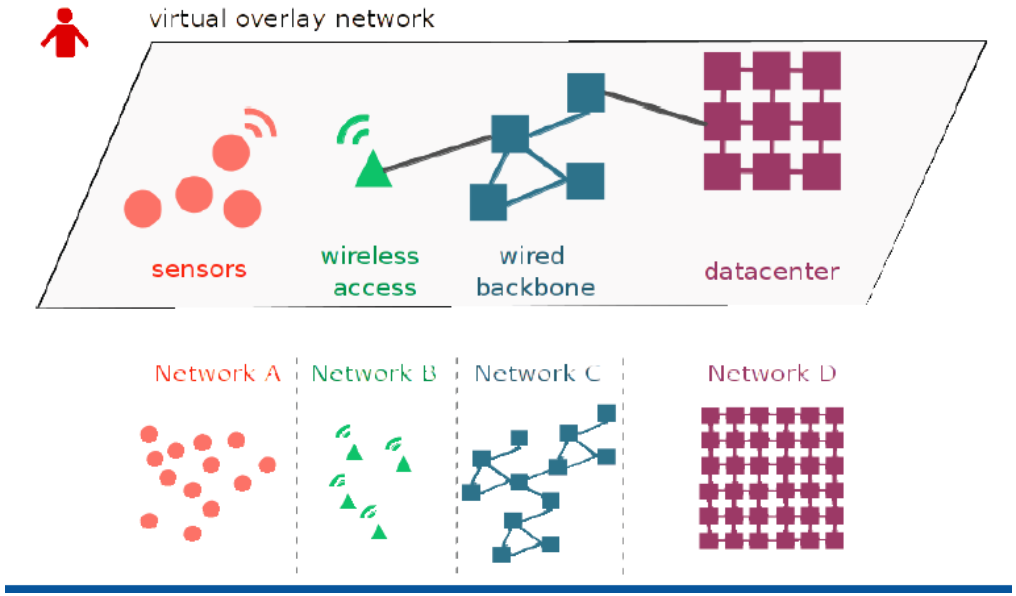
Jordan Auge, Loic Baron, UPMC, France

Peter Becue, Brecht Vermeuken, iMinds, Belgium

Sandor Laki, Jozsef Steger, ELTE, Hungary

The vision

- Originated in 2005



FIRE

Three main technology accelerators:
- Virtualization,
- Open Source,
- Open Data.

Enabling OpenLab vision

- Considerations about technical, legal, managerial and commercial enablers to achieve this vision
  - What is the right level of abstraction, the **minimum set of functionalities** to be adopted to share resources owned by various authorities?
  - How can we best support the **experimental life-cycle**
  - What is the **governance** model that best supports subsidiarity?
  - What is must have for **Operation** and **Support**
  - and finally, is there a **business** model or how can we contribute to sustainability?

A vision comes true

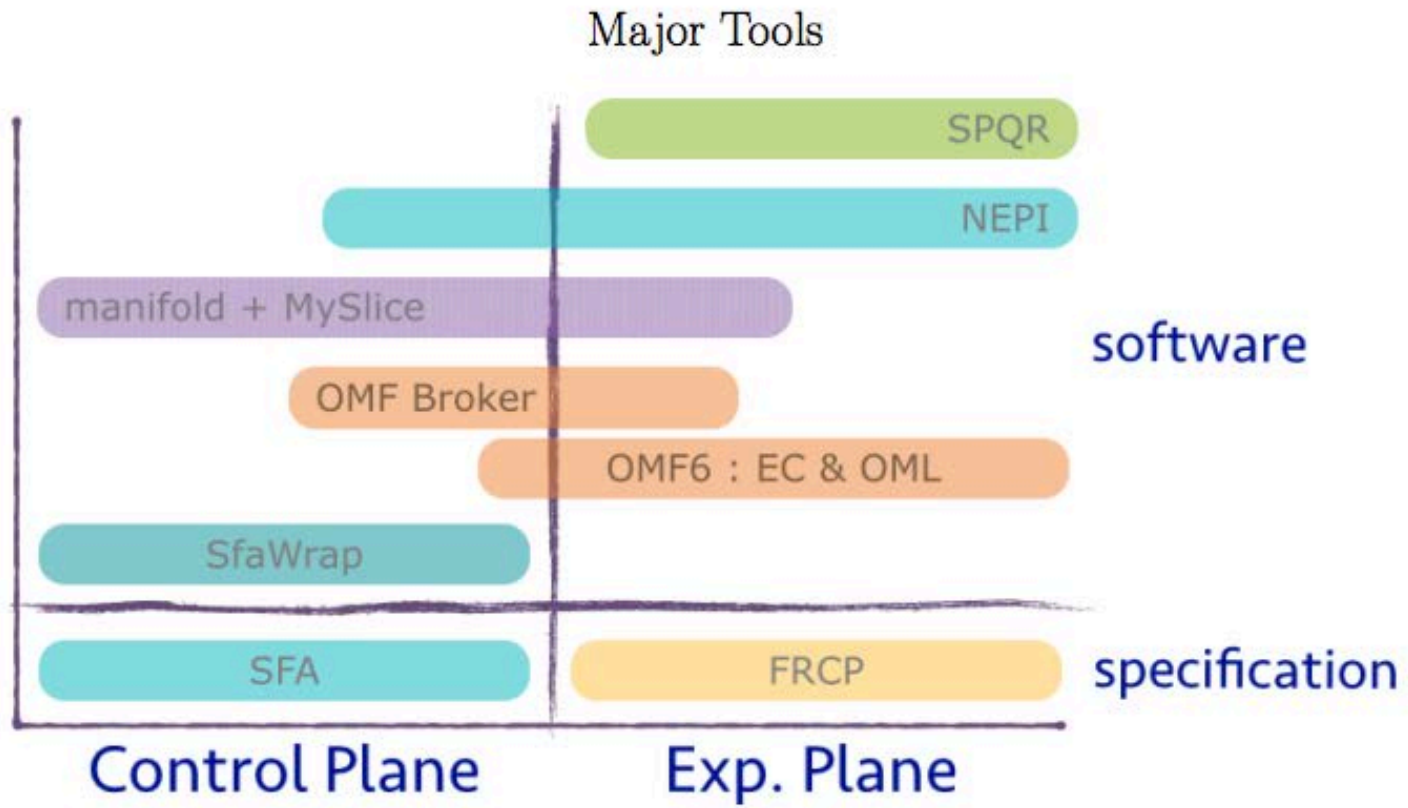
- The OneLab Facility
  - The purpose was to clearly **separate** the OneLab facility from the project that funded most of the effort produced to enable **OneLab**, namely **OpenLab**
  - An **Internet of Testbeds** can be organized with an incremental growth
  - **Heterogeneity** opens new and modern research avenues that cannot be served by the current testbed offering
  - **Authorities** can join the OneLab federation and become stakeholders of the global Facility
  - **Experimenters** have to register with an authority or directly with the OneLab organization that now has a legal existence
  - OneLab is providing **various services** (first line support, monitoring of resources, the handling of the user's registration and possibly the dissemination and outreach)

Testbed abstractions

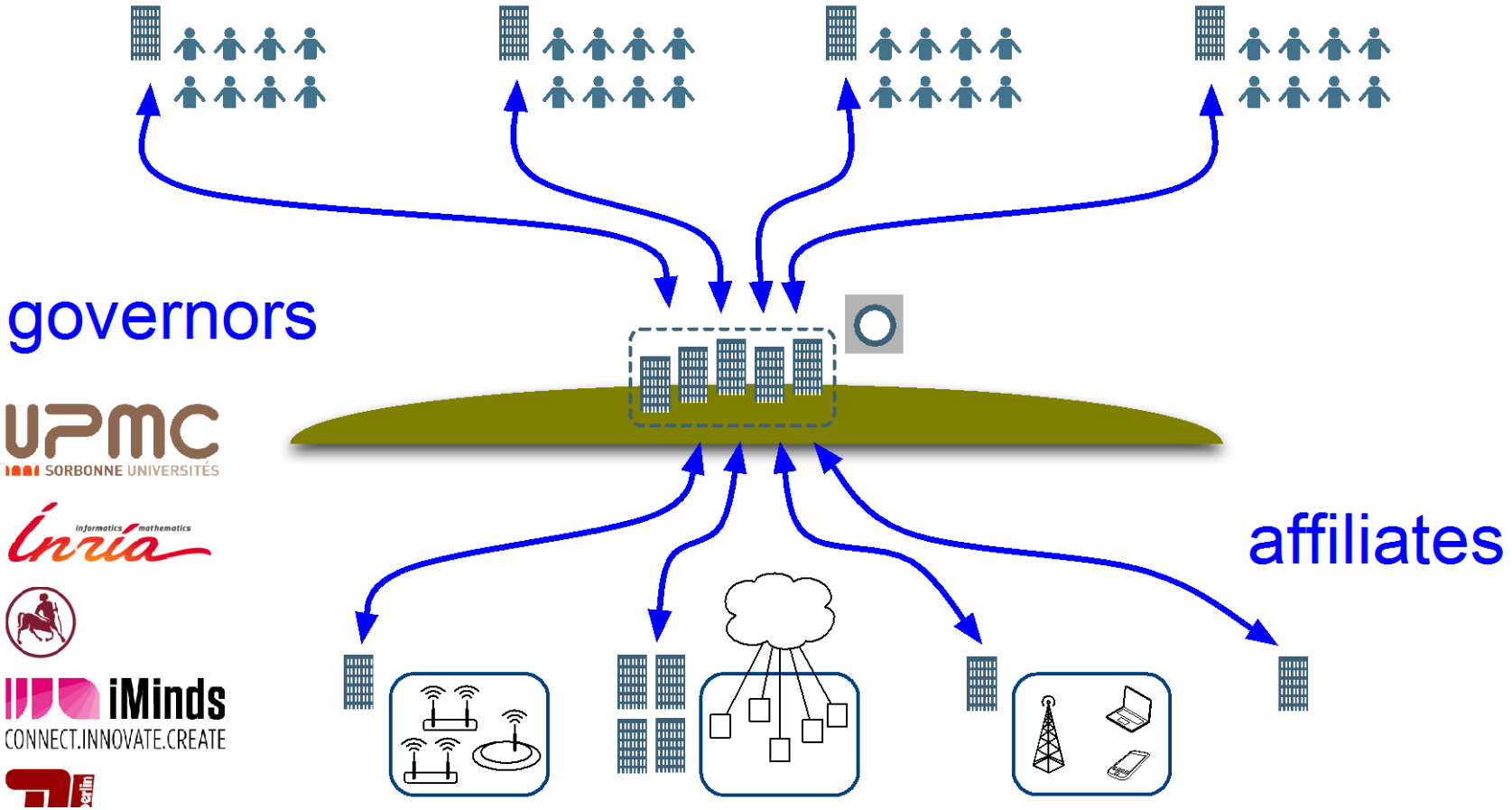
	object	service
	**resource**	Testbed ensures proper management of nodes, links, switches, …
	**user**	Testbed guarantees the identity of its users
	slice	A distributed container in which ressources are shared : • sharing with VMs, in time, frequency, within flowspace, etc. The base for accountability
	**authority**	An entity responsible for a subset of services (resources, users, slices, etc.)

Delivering to the community

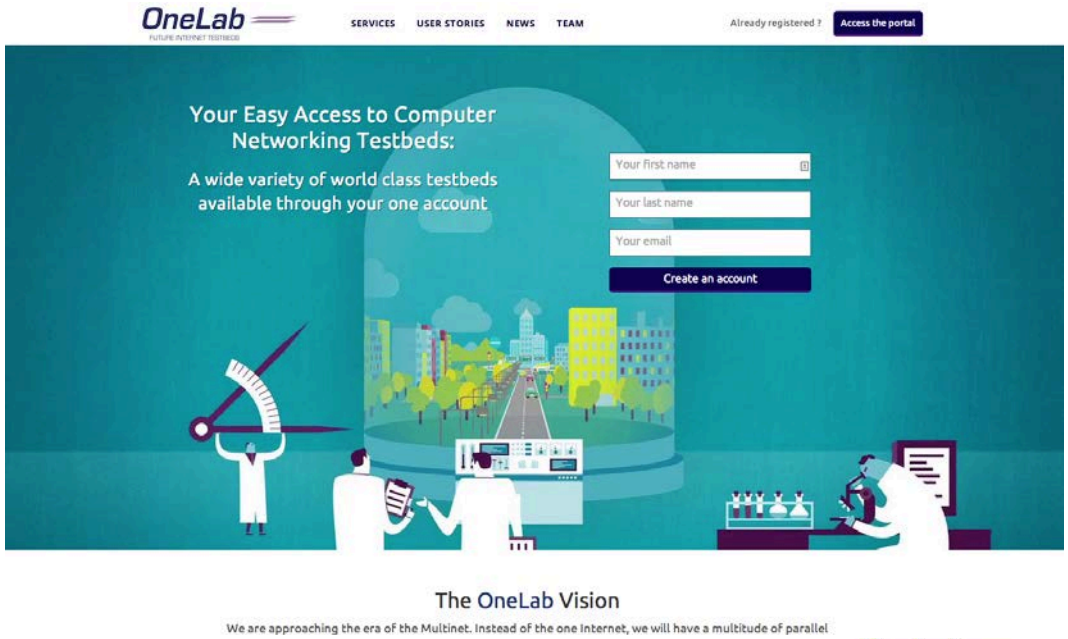
A summary of the tools contributed by OpenLab

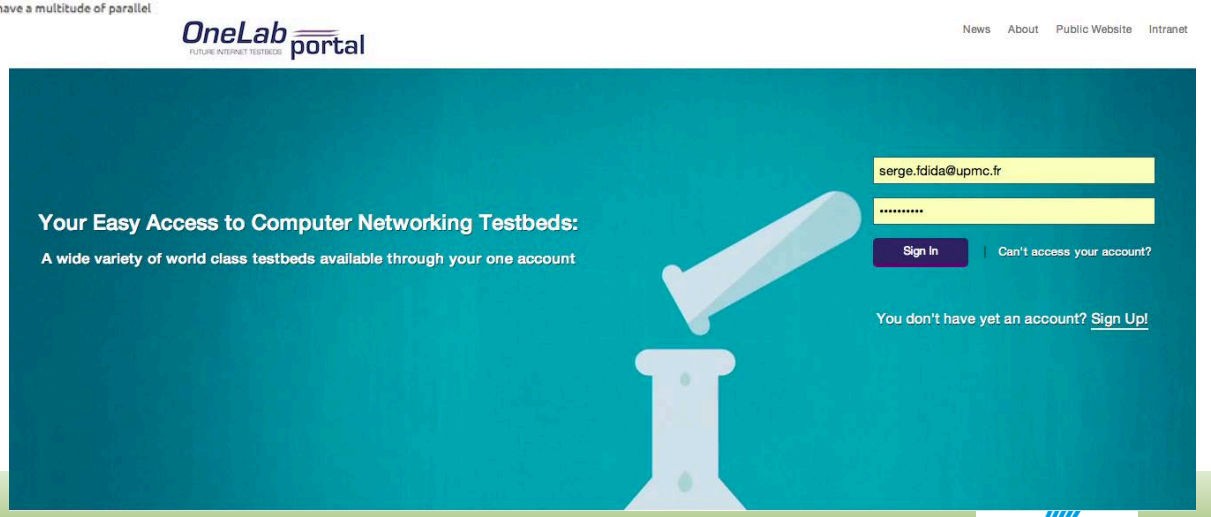


OneLab Governance & Legal Framework

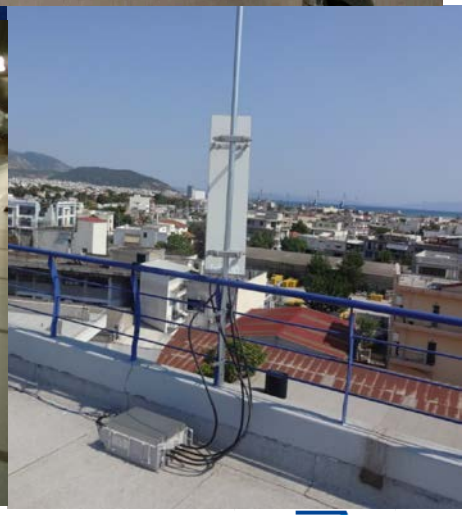


OneLab Web site and Portal

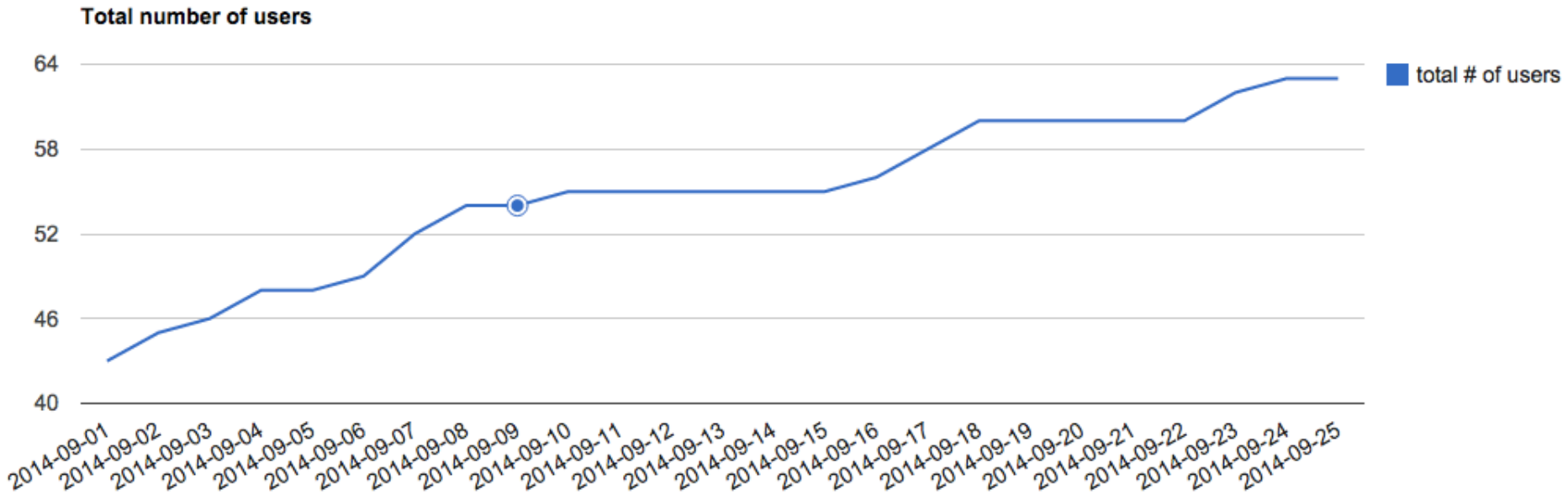




The OneLab NOC



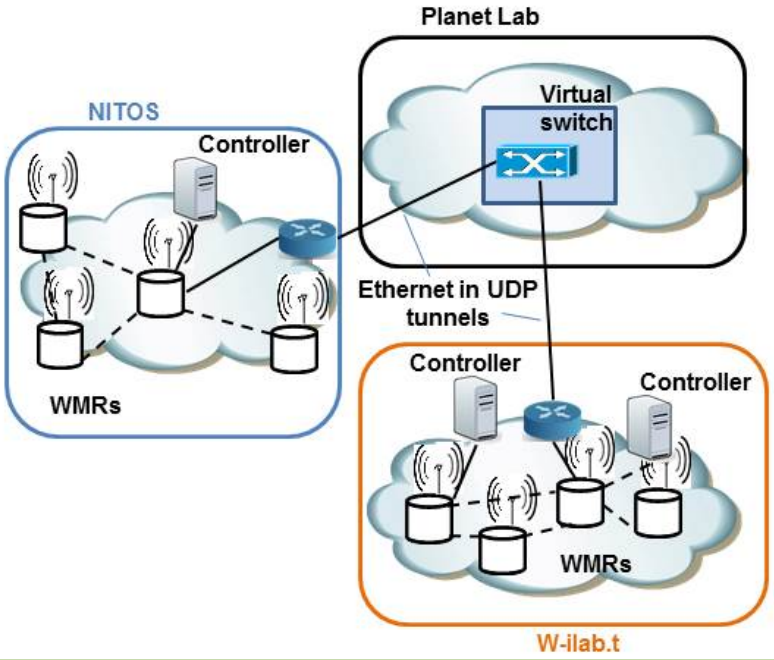
OneLab usage



OneLab facility opened on August 22, 2014

What is OneLab good at?

- Can we evidence the added-value for experimenters?
- The Express experiment as an illustration of OneLab capabilities
- Integrated demo: A full experiment lifecycle through the OneLab portal (this publication)



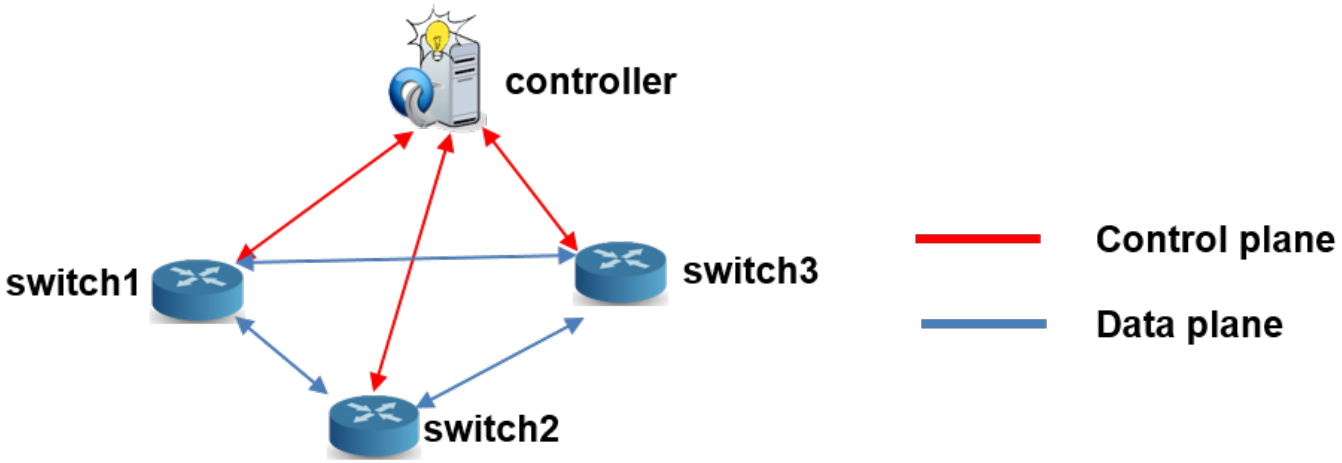
Objectives of the EXPRESS integrated experiment

- Design an innovative, *resilient Software Defined Network (SDN) systems* in order to extend the SDN applicability from fixed networks *to intermittently connected networks*, like *wireless mesh networks*.

- *The need:* After simulation based evaluation of the scheme, we need *to measure the performance of the proposed scheme in real conditions*:
  - Evaluate its feasibility when it is implemented in real devices
  - Measure actual delays in a real wide-area network that spans in Europe

SDN Background

- An SDN switch is connected to an SDN controller from where it receives the forwarding rules.

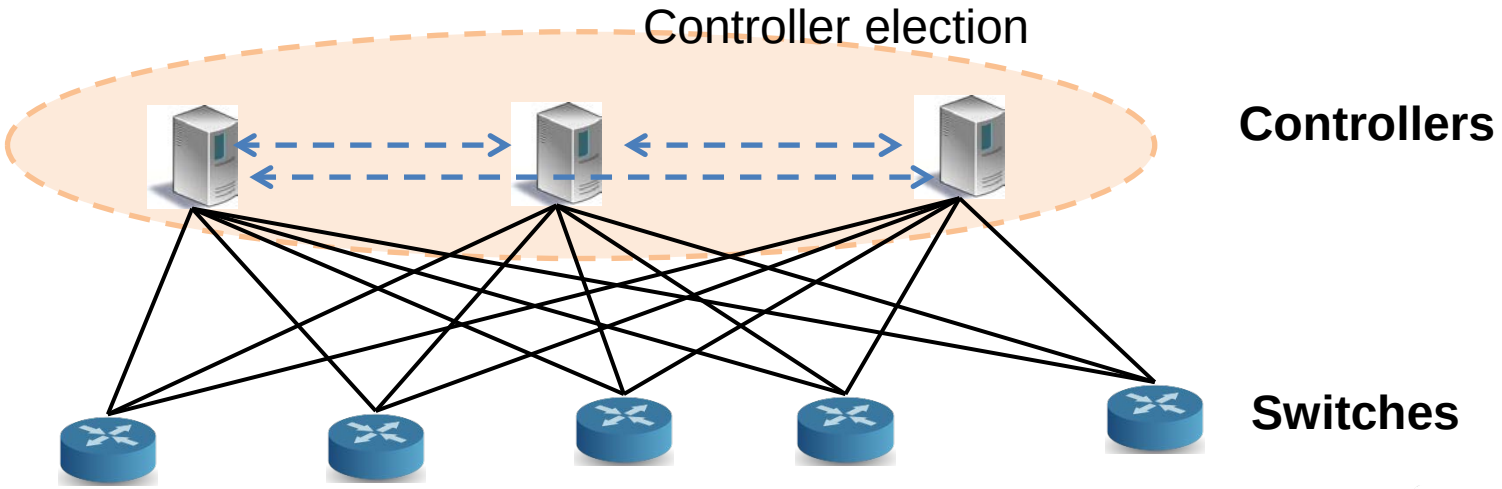


- Every time a new packet arrives to the router and there is no entry in the flow table for it, the router communicates with the controller.

SDN Background

Several approaches for assigning controllers to routers:

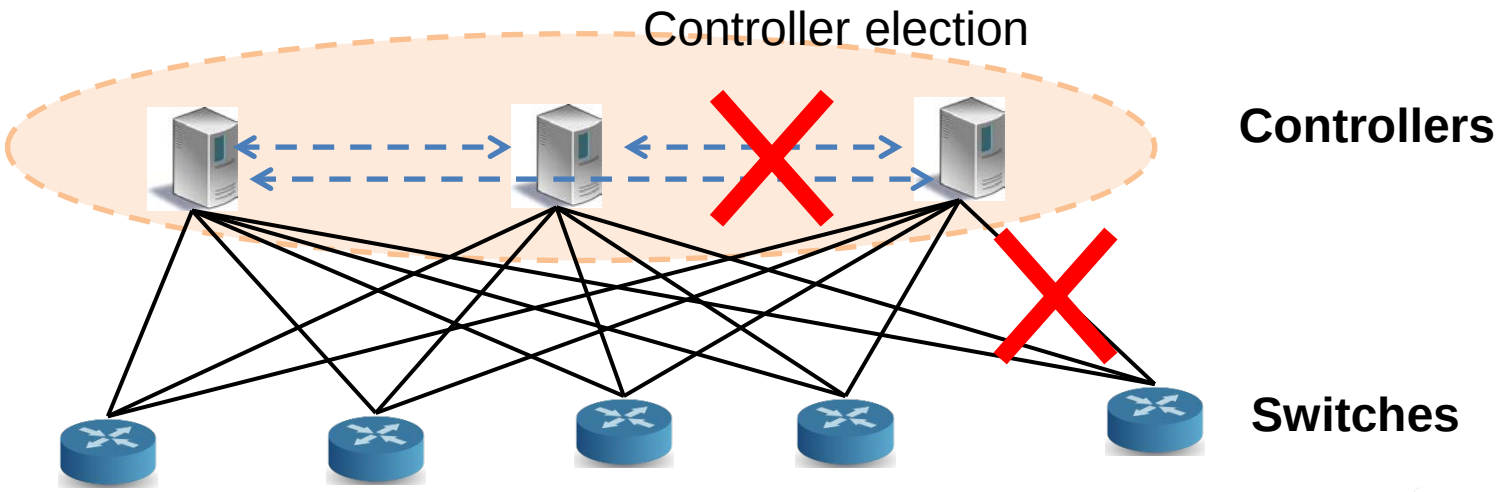
- Static assignment
- Dynamic assignment (Election process among controllers)
  - Process:
    - Communication between the controllers
    - Election of the high priority controller
    - Distribution of this info to the routers



SDN Background

Problem of these approaches when vulnerable links exist:

- If a connection between the controllers is lost, the election process can not be finalized
- If a connection between a switch and a controller is lost, the switch can not be informed for the new elected controller



Scientific questions: The goal of Express application of SDN to Wireless Mesh Networks

- To *build an efficient mechanism for enabling communication of switches to the related controllers:*
  - *It should* enable *resilient communication* between switches and controllers in environments with *vulnerable links*
  - It should work well in Wireless Mesh Networks
  - It should be distributed
  - It should be able to adjust dynamically to the network changes

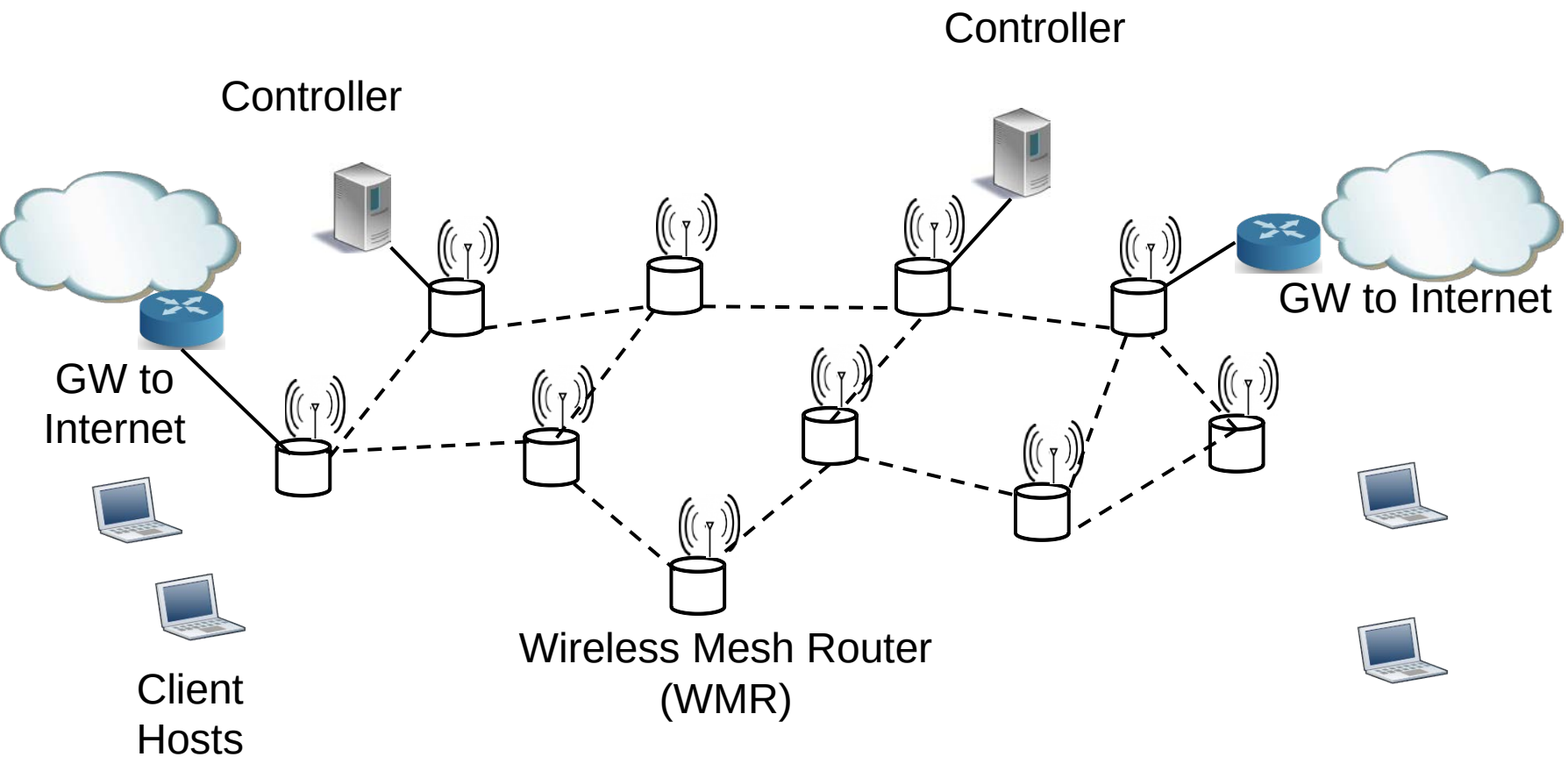
The Proposed Mechanism

The proposed scheme is based on a selection process that is driven by the switches:

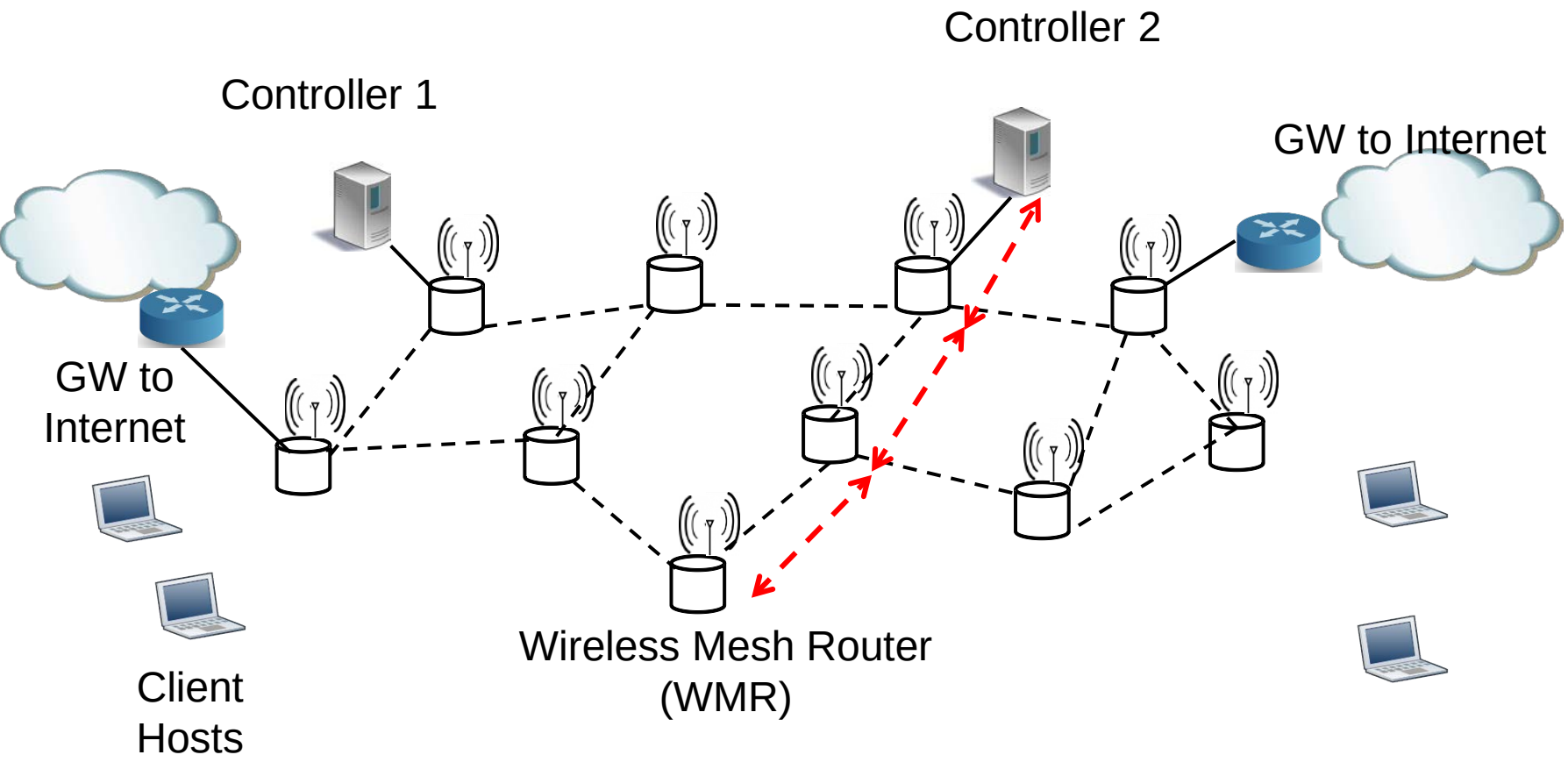
- Available controllers are announced in the mesh network
- Each switch runs a distributed algorithm in order to choose the best controller
- Once there is a break in the connection of the switch with the respected controller, the switch chooses a new controller based on the selection algorithm and the available controllers.

- Possible Selection approaches:
  - Fastest link between the switch and the controller
  - Most reliable link between the switch and the controller

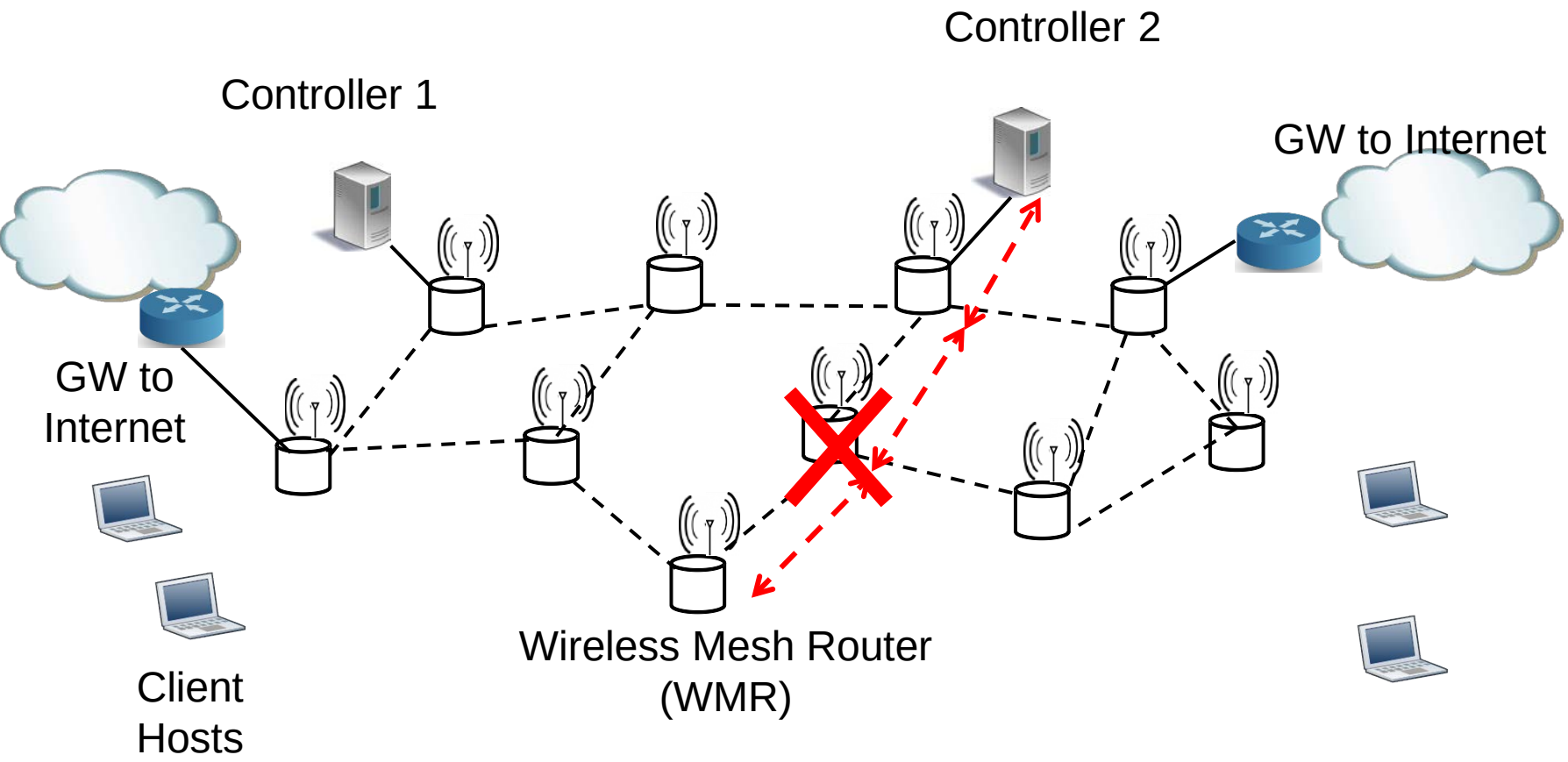
Setting the scenario
wireless mesh Software Defined Network



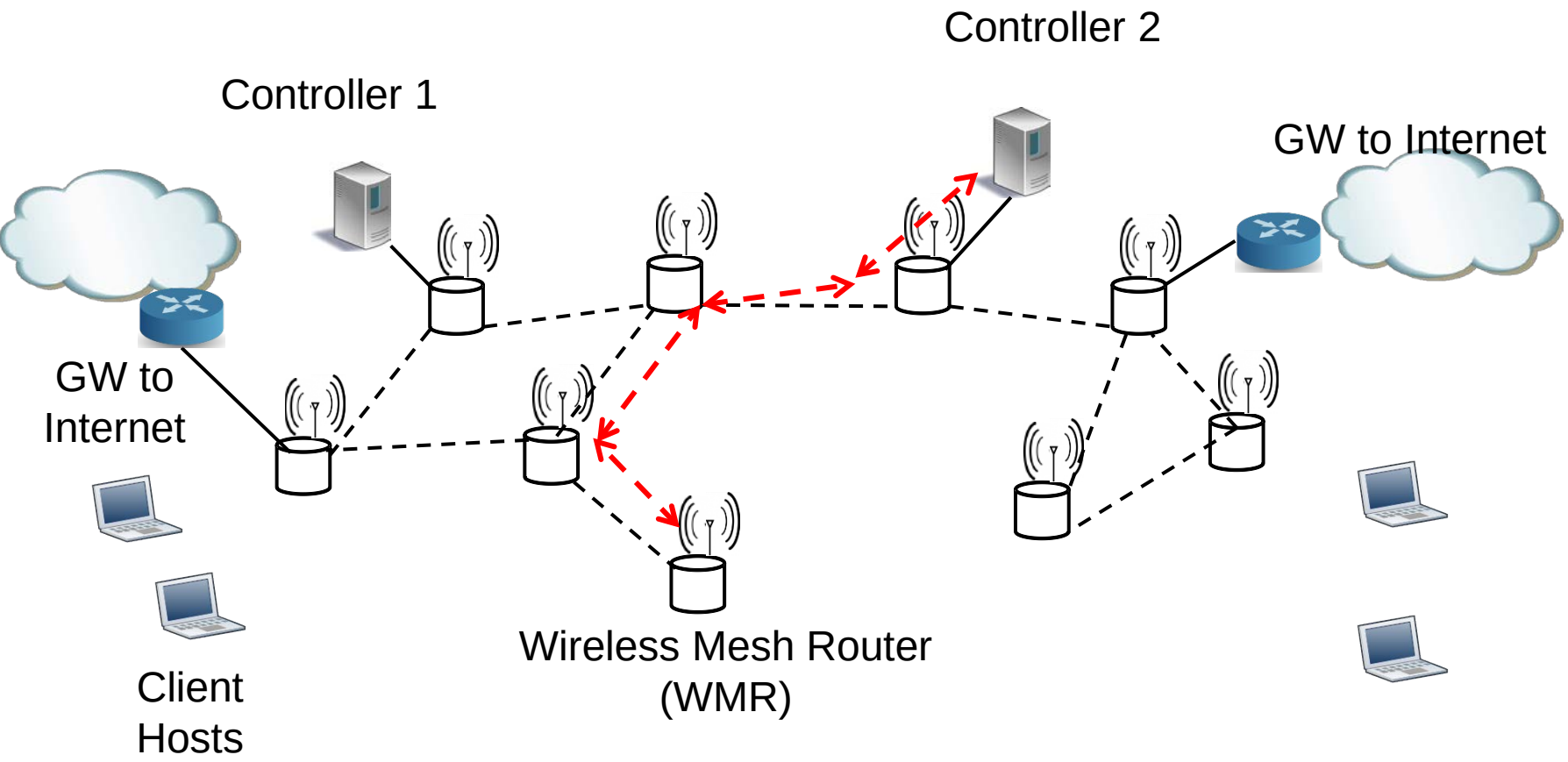
An illustration of the EXPRESS mechanism



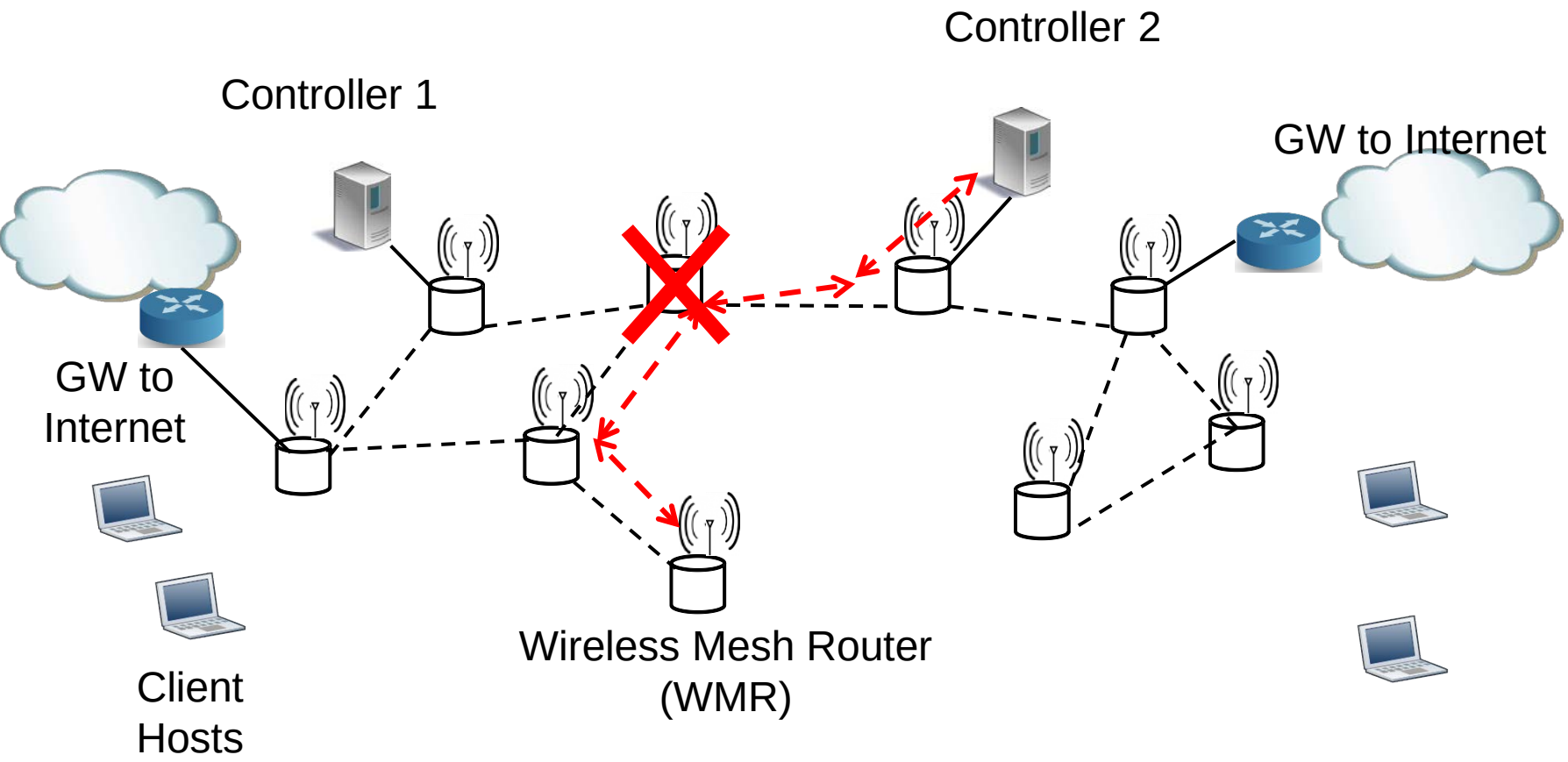
An illustration of the EXPRESS mechanism



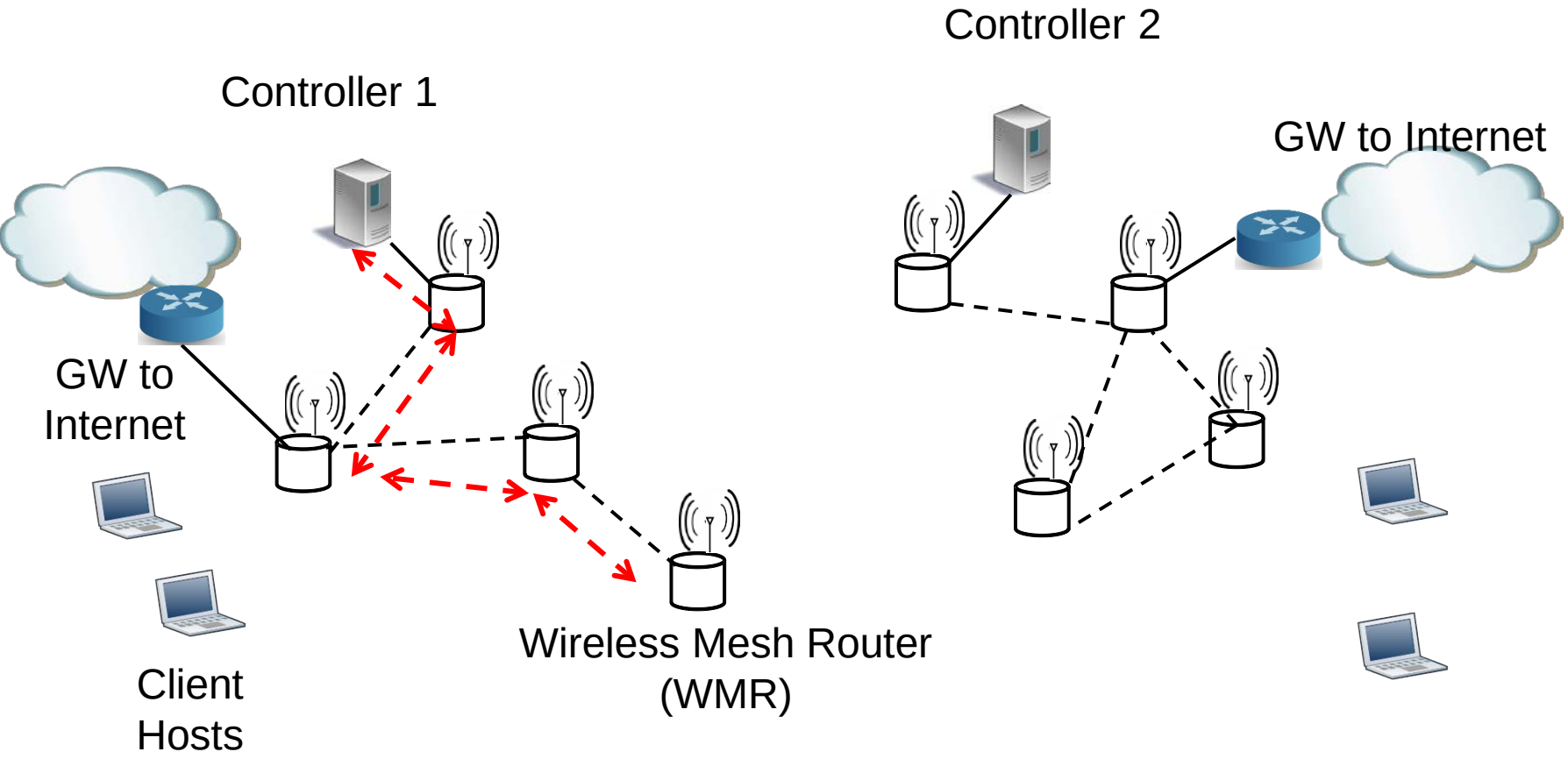
An illustration of the EXPRESS mechanism



An illustration of the EXPRESS mechanism

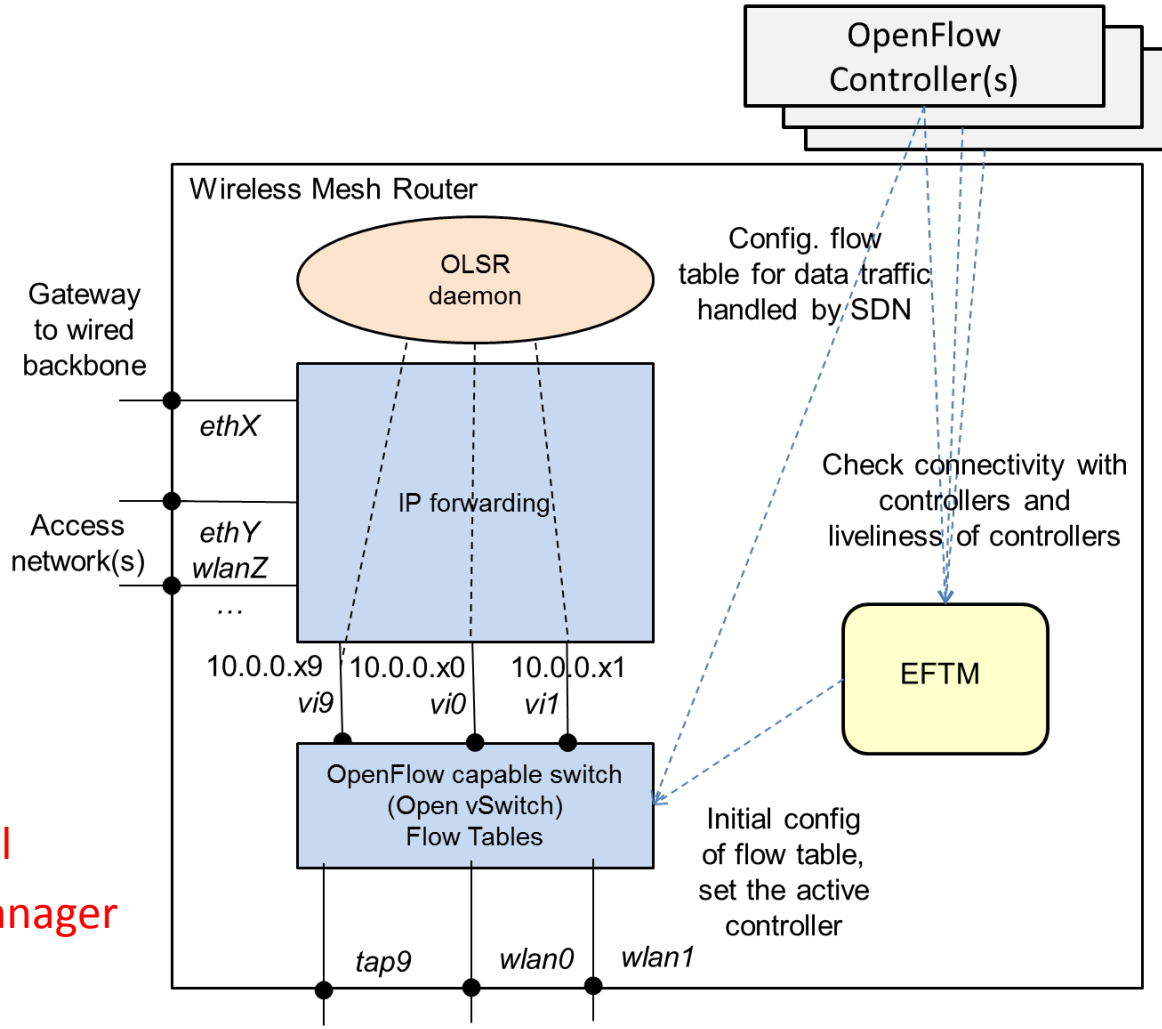


An illustration of the EXPRESS mechanism



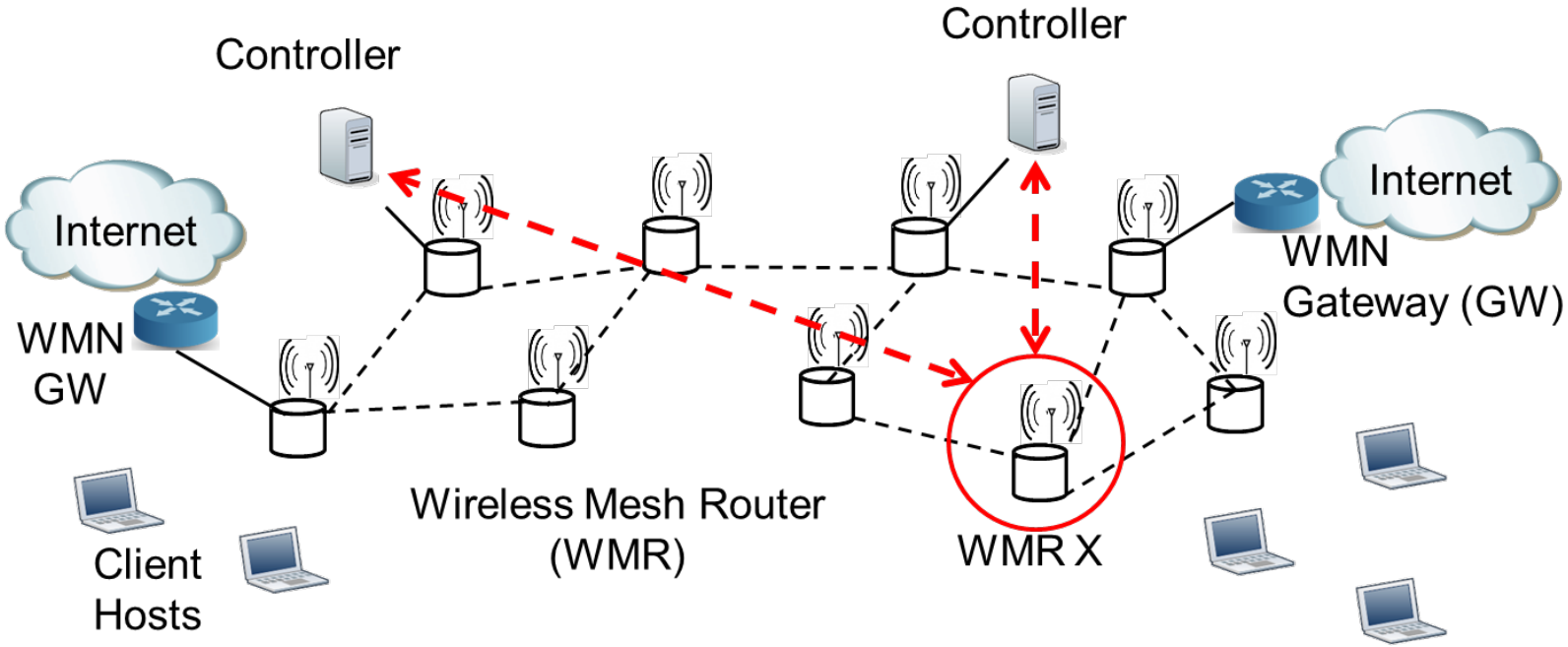
Solutions to 1st challenge (SDN in WMNs)

- Combination of:
  - OLSR based IP forwarding
  - SDN based rules
- "In-band" SDN control plane exploiting plain IP forwarding / OLSR routing
  - OLSR: Ad Hoc routing protocol
  - EFTM: External Flow Table Manager



Solutions to 2nd challenge (SDN controller selection)

- From master *election* to controller *selection*



- A WMR is in charge of selecting the preferred controller among those available in its portion of the network

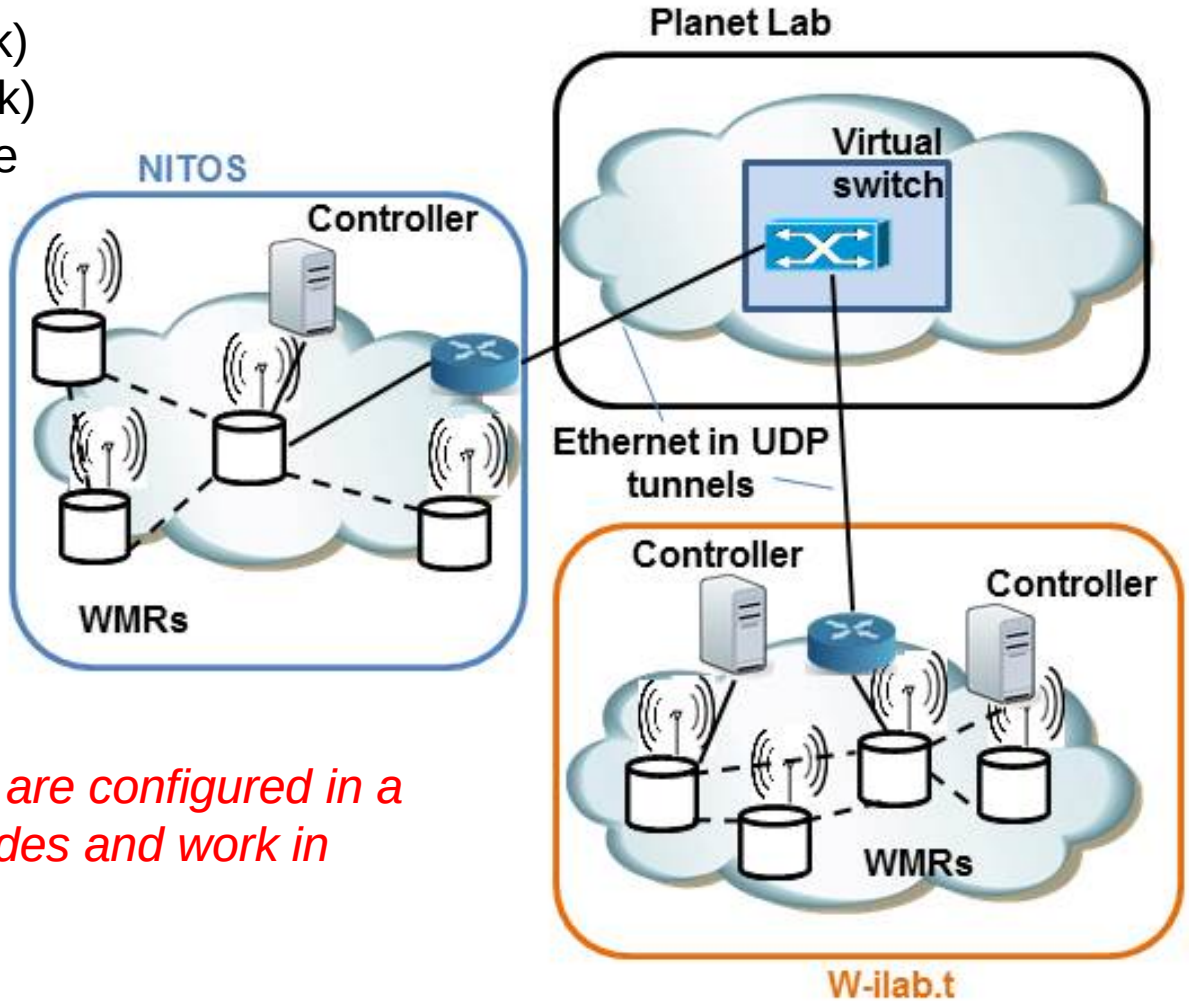
Advantages or the proposed controller selection approach

- At any given time, a WMR is connected only to a controller: no conflicting rules can be injected
- There is no need of a coordination mechanisms among controllers with strict real-time requirements (obviously all controllers should follow the same service logic…)

Experimenting in OpenLab testbeds

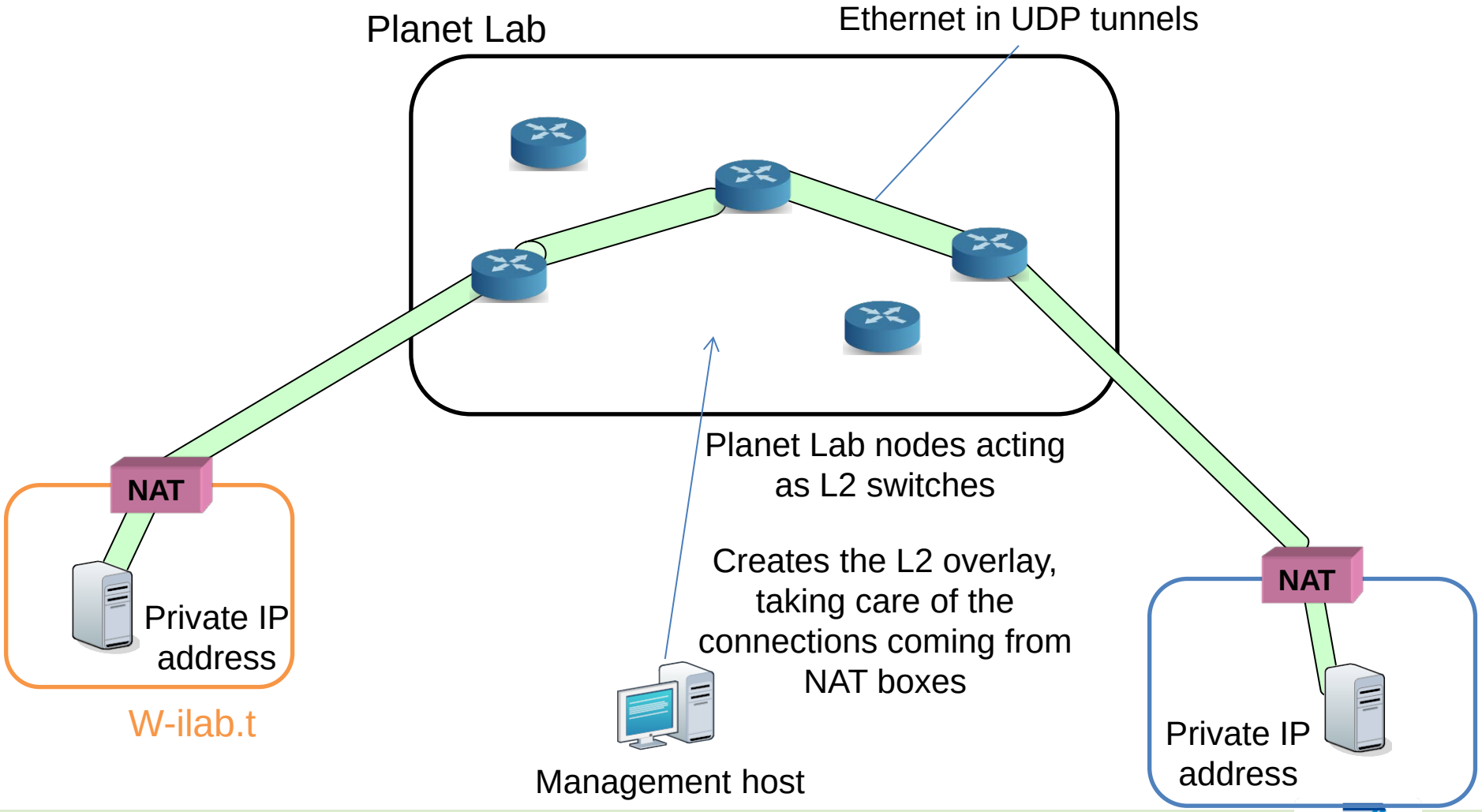
Three Openlab Testbeds:

- NITOS (wireless mesh network)
- w.Ilab.t (wireless mesh network)
- Planetlab EU (as the backbone connecting the two networks)



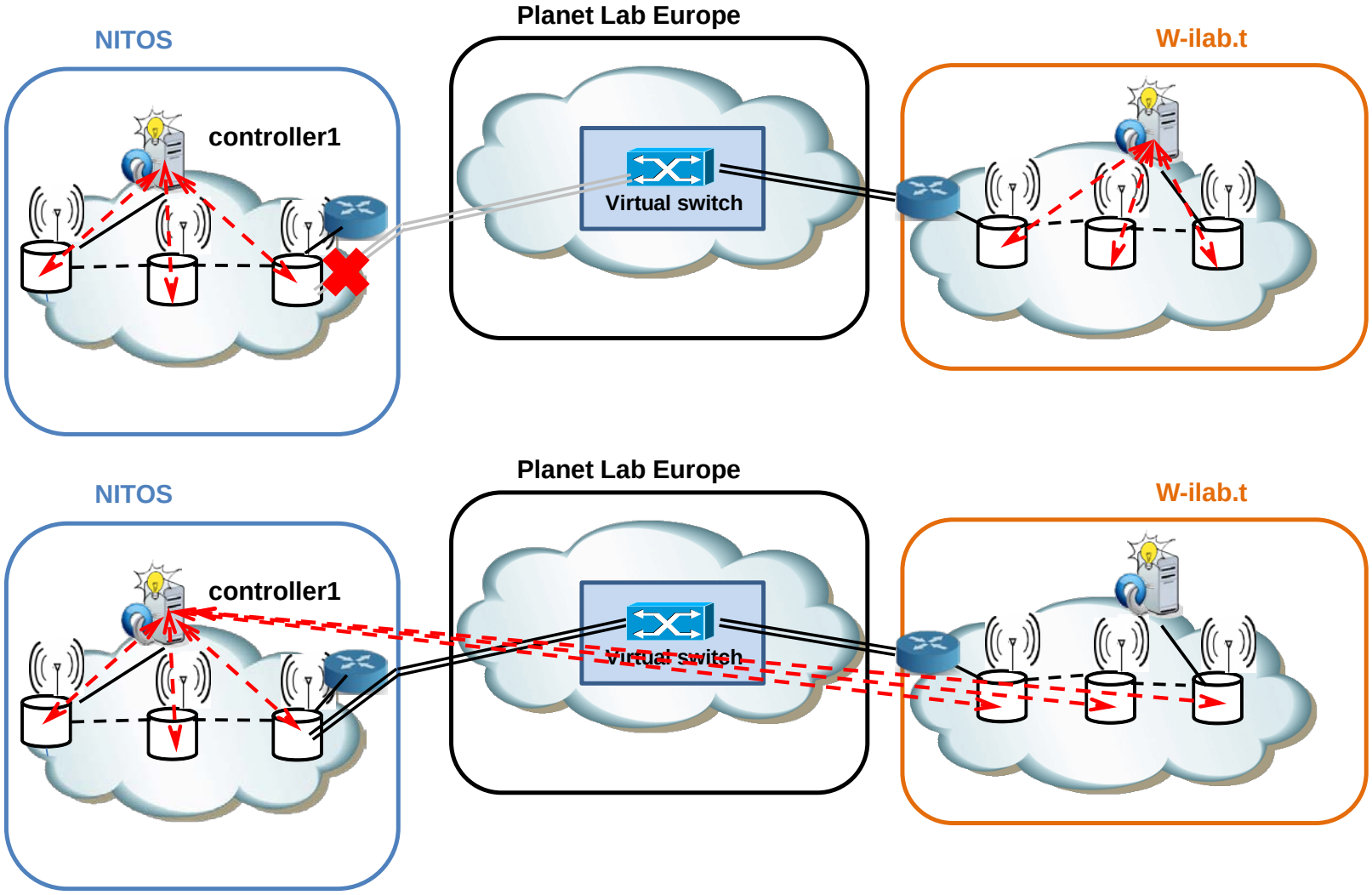
Nodes from NITOS and w.Ilab.t are configured in a way to act as wireless mesh nodes and work in parallel as OpenFlow switches

Solutions to testbeds interconnection issues

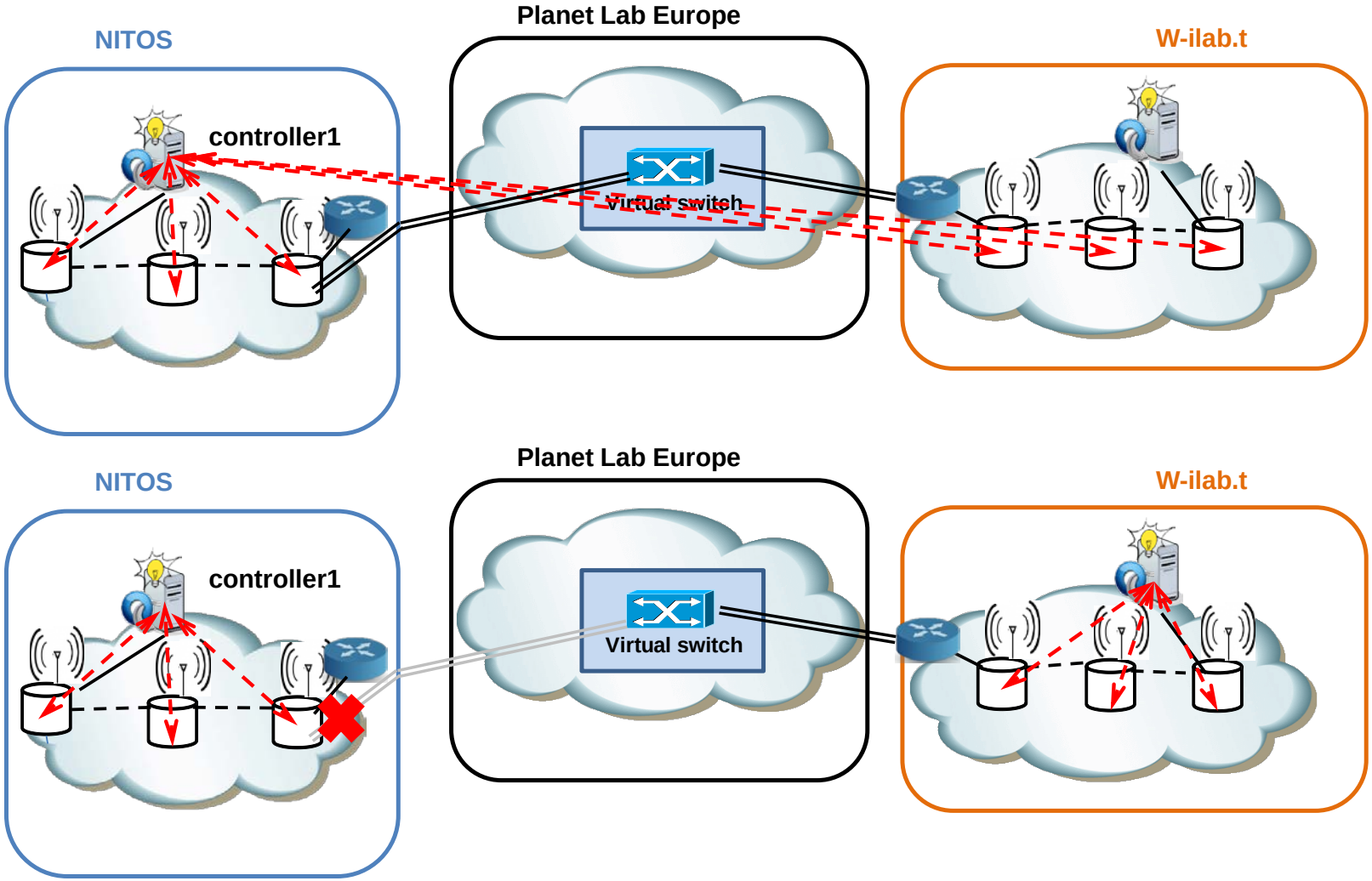


Description of the experiments

1. Network merging



Description of the experiments
2. Network partitioning

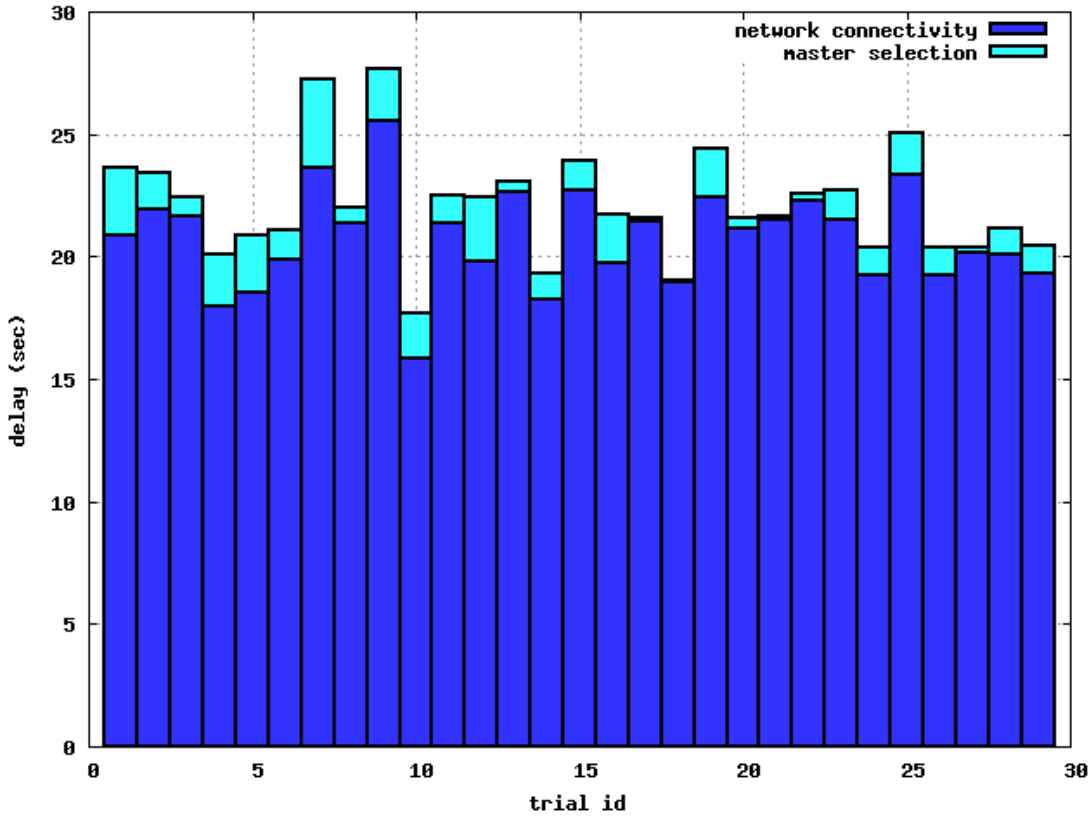


Measured metrics in the experiments

1. Network merging experiment:
   - evaluate the time needed for the WMRs to connect to a higher priority controller after the merging of two network partitions

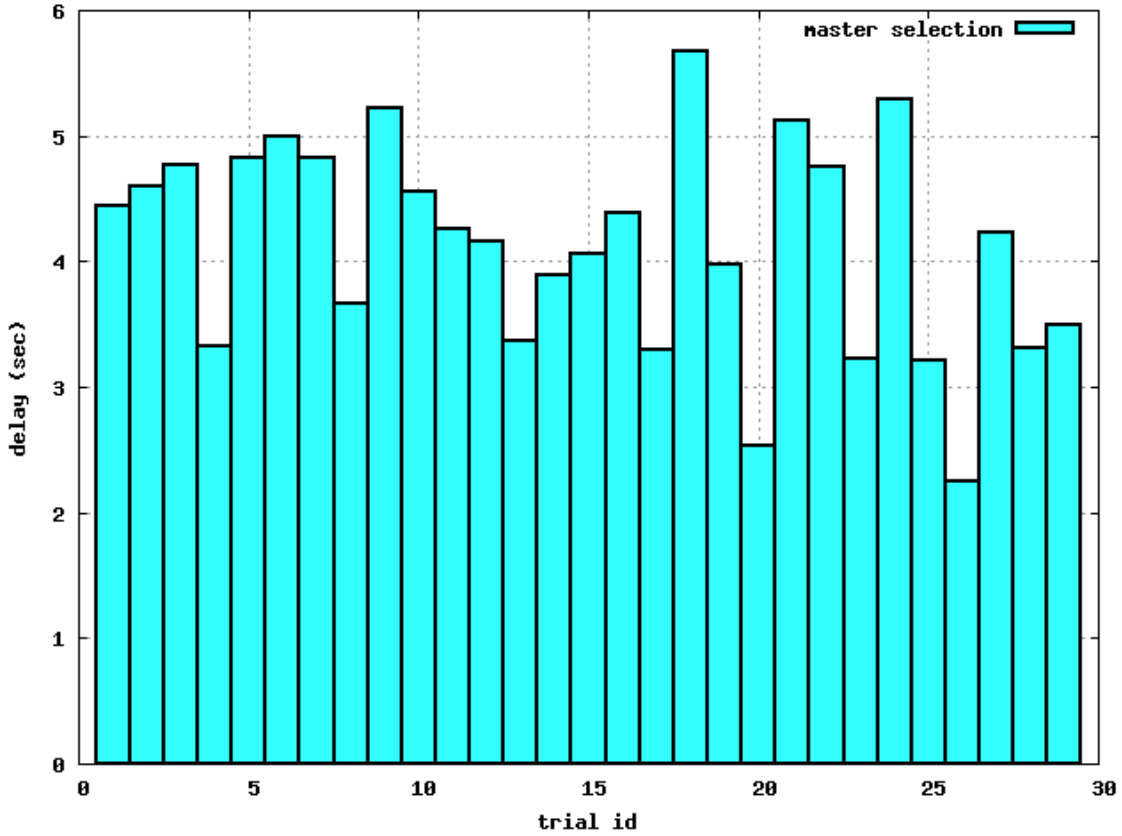
2. Network partitioning experiment:
   - evaluate the time needed for the WMRs to connect to an available controller after the partitioning the network

1. Network merging experiment



- Evaluates the time needed by the last WMR to disconnect from controller1 and connect to controller2
- Blue: time needed for the update of routing tables (topology)
- Light blue: total time to connect to controller2 (selection)

2. Network partitioning experiment

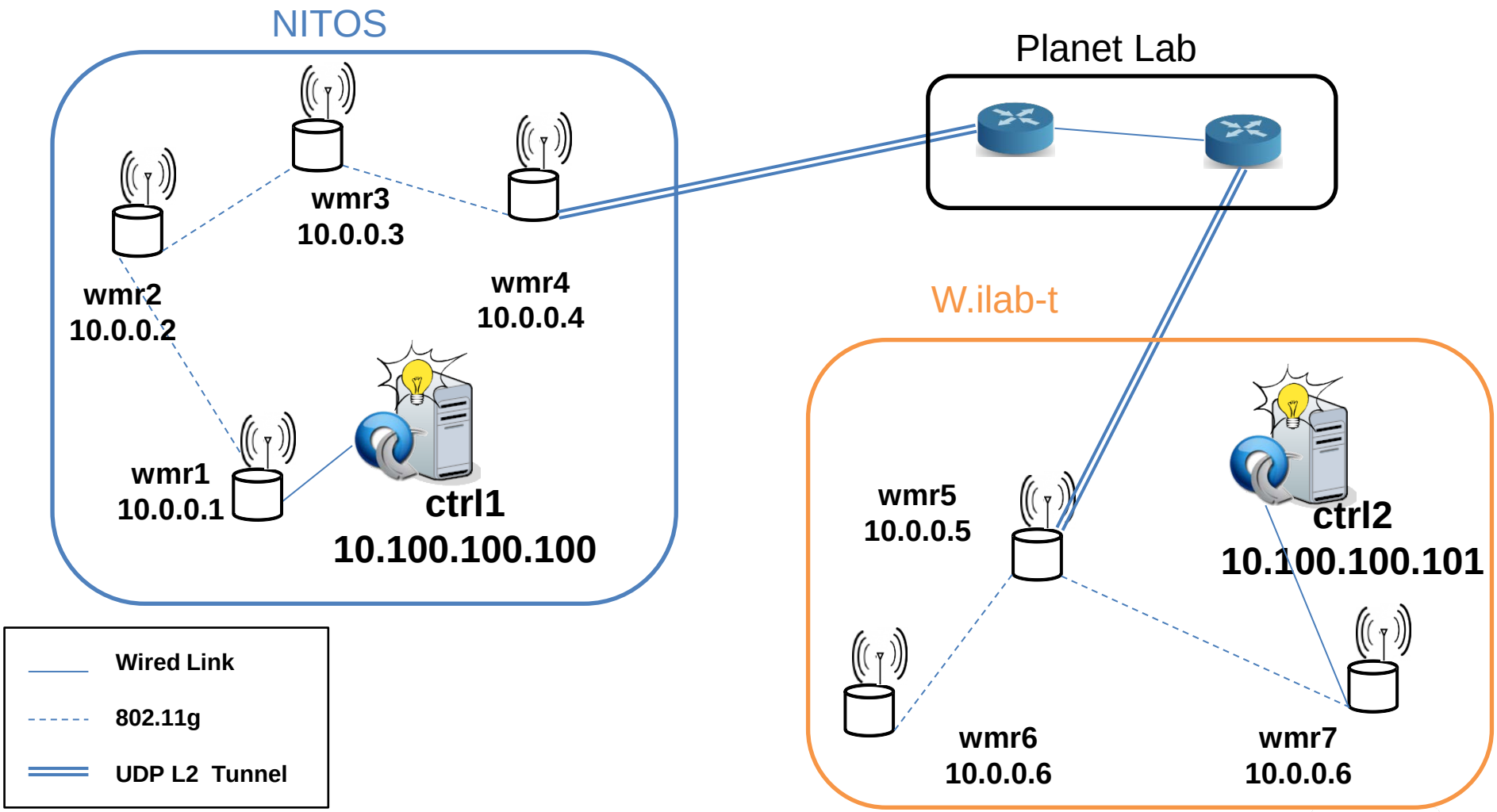


- Evaluates the time needed by the last WMR to disconnect from controller2 and connect to controller2
- Light Blue : total time to connect to controller1

Experimenting in OpenLab testbeds

1. Reservation of the resources through *mySlice Portal*
2. Development of the experiment script through *OMF*
3. Execution of the experiment through a single script
4. Collection of the measurements through *OML*
5. Visualization of results though nmVO

Experimenting in OpenLab testbeds



Demonstration

- Video

Conclusions – Scientific questions

1) We demonstrated the feasibility of the proposed SDN solution with “In-band” control plane and with the co-existence of plain IP routing and SDN based forwarding

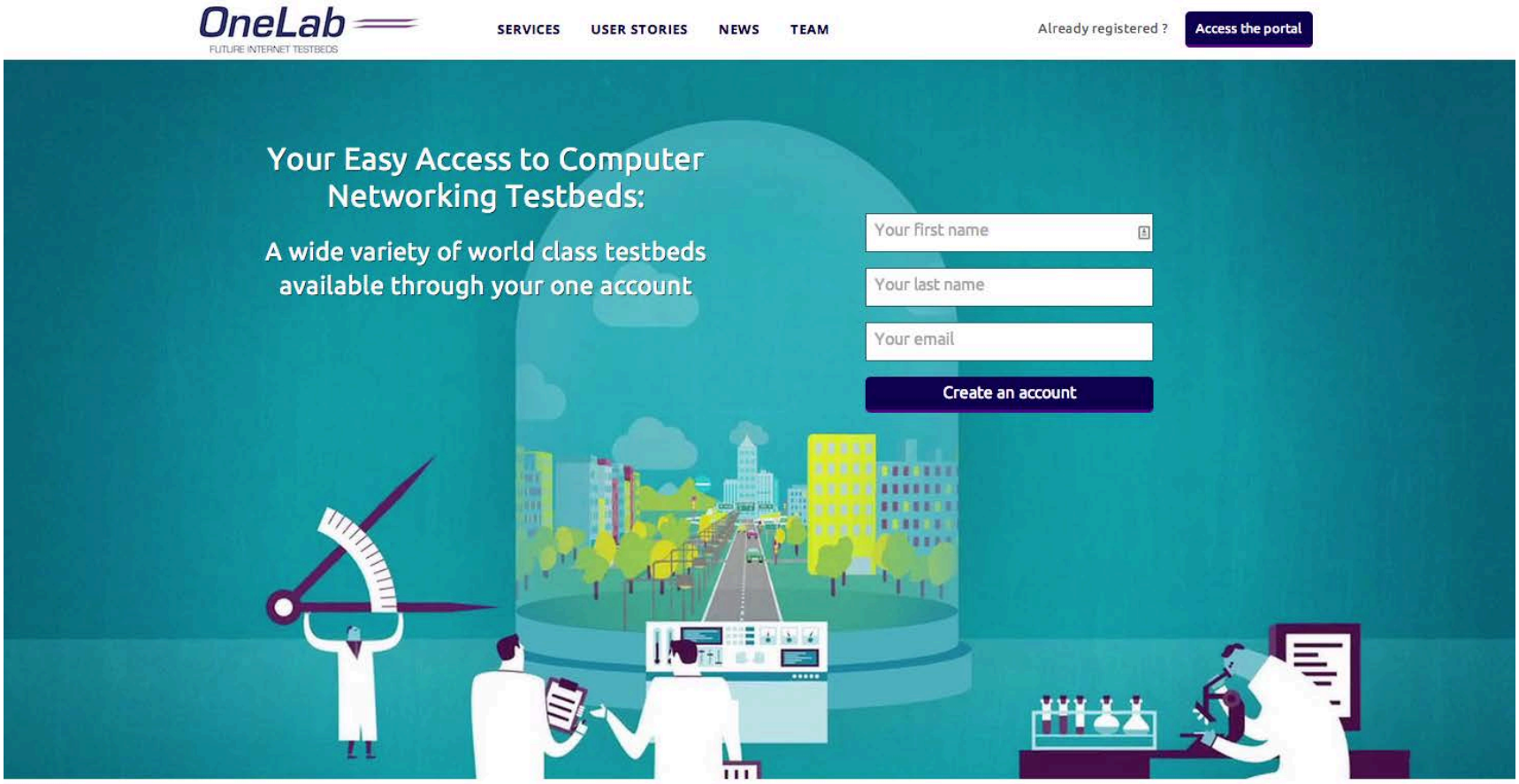
2) Controller selection approach: the time scale of operations is in the same order of the underlying OLSR based restoration time

3) A SDN based approach for the control of a Wireless Mesh Network proved to be feasible in a wide area experiment

Conclusions – OpenLab testbed

- The EXPRESS experiment involved three different and heterogeneous testbeds.
- A first version of the experiment was conducted with a lot of “manual” configuration and analysis
- A second version of the experiment was conducted with a reduced effort thanks to the tools provided by the OpenLab facility:
  - One Lab / my slice single sign on
  - Single OMF script to control multiple testbeds
  - Database and visualization tools

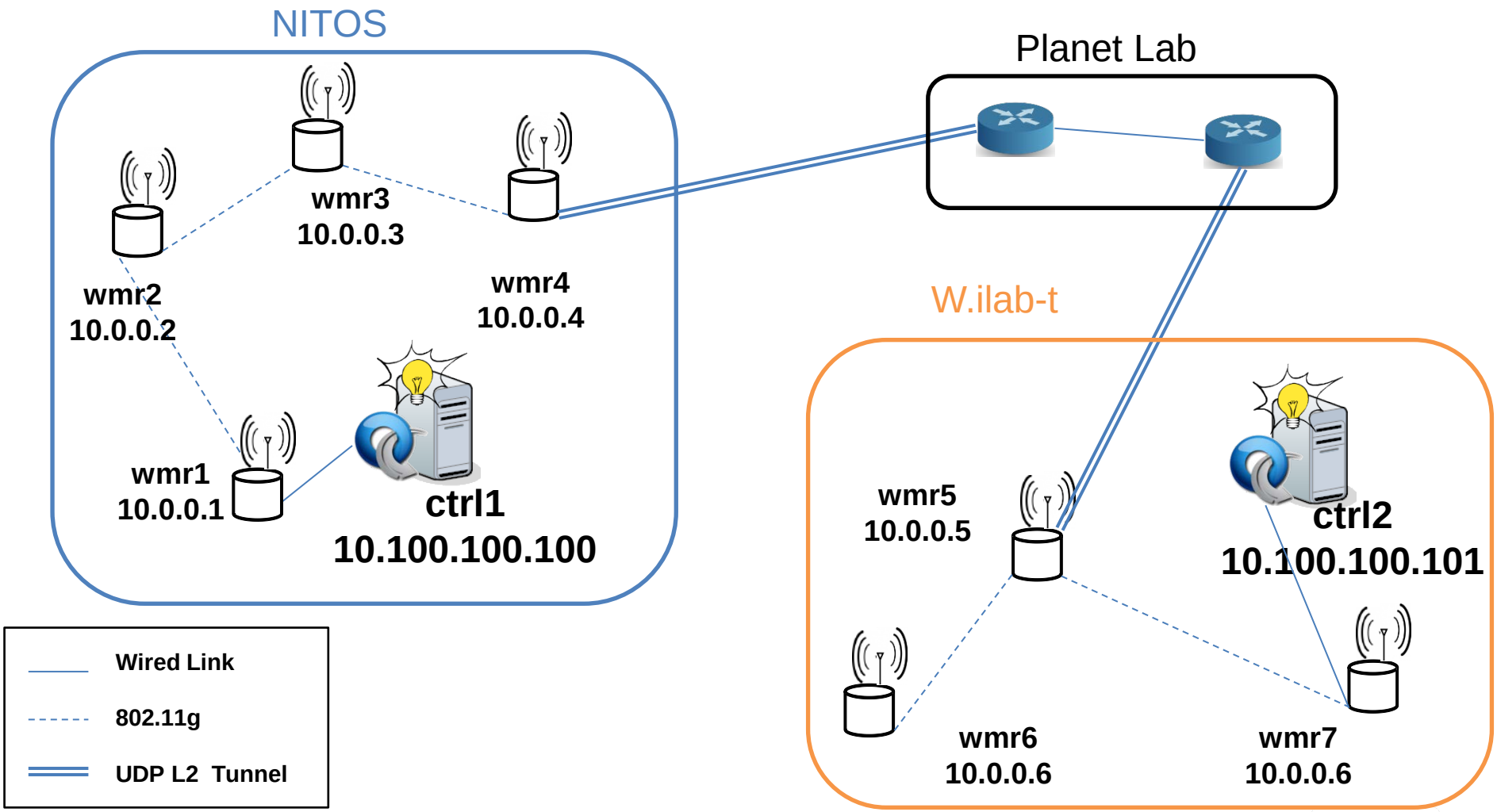
Questions?



www.onelab.eu **contact@onelab.eu**

BACKUP SLIDES

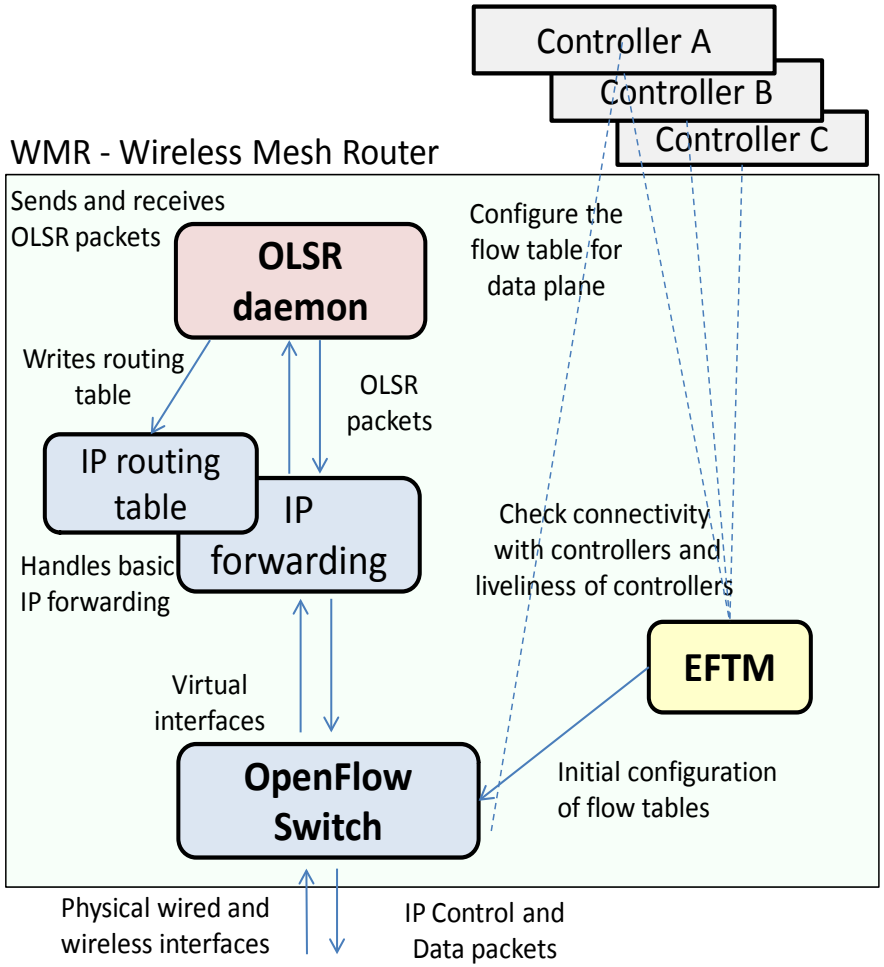
Experimenting in OpenLab testbeds



Wireless Mesh Router configuration

- OLSR Daemon
- Open vSwitch
- EFTM - External Flow Table Manager

- WMR can be easily configured using OMF



NITOS experiment details

- Recources :
  - 7 Grid nodes
  - Experiment Description (part_test.rb)
  - express OS image
- Nodes' details:
  - WMRs nodes {wmr1, wmr2, wmr3, wmr4, wmr5}
    - connected to the same ad hoc network (10.0.0.0/24)
  - WMRs directly connected to a controller {n1,n5}
    - - wired interface to the controller (ex 10.100.101.2)
  - Controllers {ctrl1,ctrl2}
    - wired interface (ex 10.100.101.100)
  - Priority of ctrl2 > ctrl1

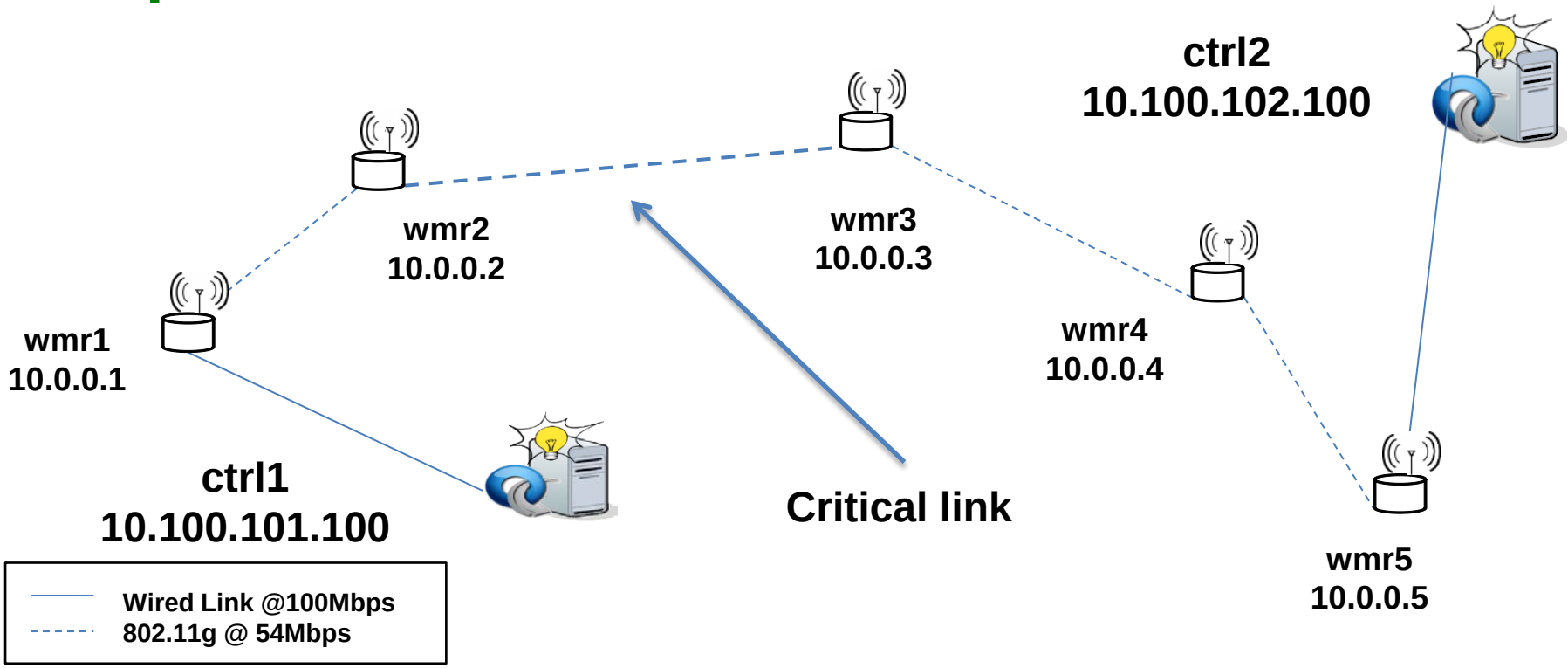
NITOS experiment details

- Applications defined in OMF Experiment Description script:
  - wmr : this script configure the wmr's interfaces and the openvSwitch bridge
  - ctr_selection : this is the External Flow Table Manager (EFTM) that performs the control selection algorithm and manage emergency conditions
  - ctr_wrapper : pox controller wrapper; run the controller, parse the output and logs events in the db
  - ovs-ofctl : utility used to simulate a link going up or down
  - tsp_wrapper : simple utility to put a timestamp in the db

NITOS experiment details

- Performed tasks :
  1. Configure all WMRs
  2. Start EFTM process on all the WMR nodes (continue in the background)
  3. Critical link up
  4. Start ctr2
  5. All WMRs are connected to ctrl2
  6. Start ctr1
  7. Critical link goes down (T1)
  8. n1 and n2 are connected to ctr1 (T2)
  9. Goto 3 and repeat for the required number of runs

Experiment in NITOS



- Initial state, critical link up
- evaluate the time needed for the WMRs to connect to an available controller after the network is partitioned