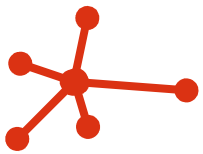
upna Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

# OBS network model for OMNeT++: A performance evaluation

F. Espina, J. Armendariz, N. Garcia, D. Morato, M. Izal, E. Magaña
felix.espina@unavarra.es



Telecommunications, Networks and Services Research Group
http://www.tlm.unavarra.es

# Introduction to the problem

- Researching on **transport of multimedia** contents on **OBS**

- **Testbeds?**
  - There are few testbeds, no one accessible to us
  - We have not enough financial resources to develop a testbed

- **Simulations**
  - There are implementations?? **Some**
  - They are accessible?? **Few**
  - Are they good enough for us?? **No**



# Introduction to the problem

- **Conclusion**: we have to **develop** our OBS simulator
- Before using the simulator...
- how good is it?

**A PERFORMANCE EVALUATION REQUIRED**



# Content

- Introduction to optical technologies: OCS-OPS-OBS
- Which simulation platform?
- OBS network model
  - Edge Node
  - Core Node
- The performance evaluation
  - Scenario and methodology
  - The effect of number of wavelengths on the OBS links
  - Comparison with INET
- Conclusions



# Content

- **Introduction to optical technologies: OCS-OPS-OBS**
- Which simulation platform?
- OBS network model
  - Edge Node
  - Core Node
- The performance evaluation
  - Scenario and methodology
  - The effect of number of wavelengths on the OBS links
  - Comparison with INET
- Conclusions



# Introduction: OCS-OPS-OBS

- **OCS: Optical Circuit-Switching**
  - Optical circuits are established (static or on demand) between routers
  - Core nodes handle passively many wavelengths => **no electronic limitations**
  - **Little flexibility**:
    - Limited traffic adaptation
    - Poor bandwidth utilization for bursty traffic
    - ...



# Introduction: OCS-OPS-OBS

- **OPS: Optical Packet Switching**
  - **Analogous to the electronic switching** of packets => ideal solution
  - Today OPS has **some serious technological limitations**:
    - **Optical queueing** is very difficult. There is not optical memory, only delay lines
    - **Switching time** of not very expensive optical switches similar to duration of optical package (millisecond – microsecond)
  - **Open question**: Will make sense OBS when we can implement OPS?



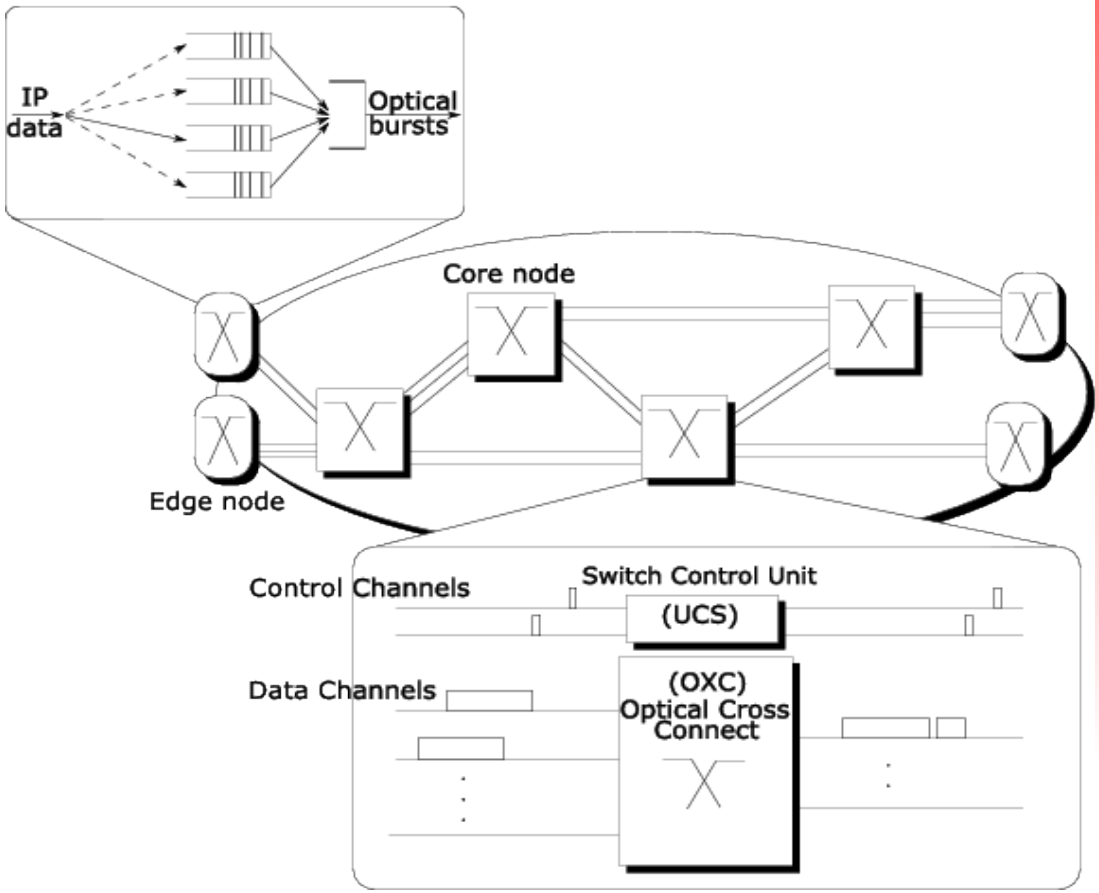
# Introduction: OCS-OPS-OBS

- **OBS: Optical Burst Switching**
  - OCS and OPS **intermediate solution**:
    - Establish a circuit only for duration of a set of packets (burst)
  - OBS **Pros**:
    - **Flexible backbone** with high bandwidth
    - **Feasible technology** (no optical buffers needed)
  - OBS **Cons**:
    - Introduces **latency** (waiting to gather packets for a burst)
    - **Losses come in bursts**



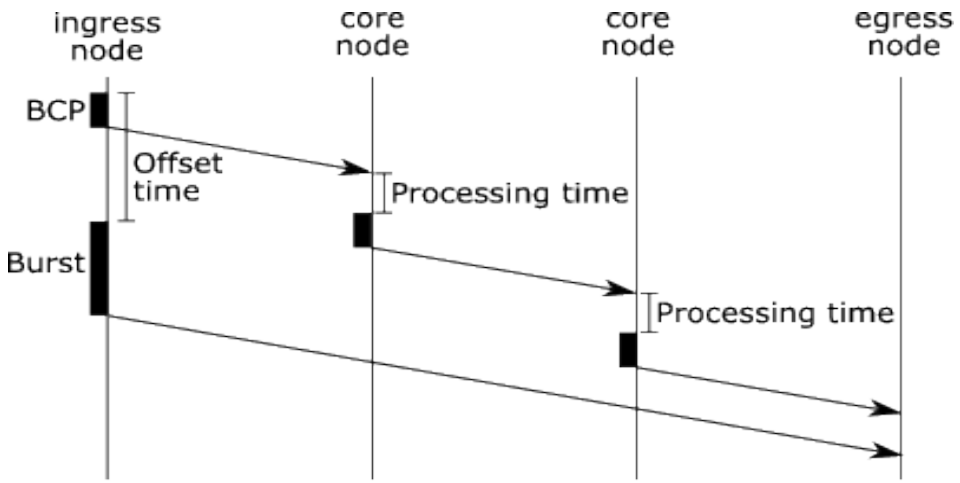
# Introduction: OCS-OPS-OBS

- **Packets buffered** at ingress edge nodes based on Forward Equivalence Classes (FEC)
- OBS FEC: **burstifiers**



# Introduction: OCS-OPS-OBS

- A **Burst Control Packet (BCP)** is created and **sent an offset time before** the burst




- BCP is **electronically switched and processed** at every backbone node
- Processed: **decide** the appropriate **forwarding** path for the associated optical burst

# Introduction: OCS-OPS-OBS

- Generally OBS uses **one-way signalling schemes initiated by source**:
  - Bursts are sent without waiting for acknowledge.
  - **Bursts** may **compete** for the same **resources** in **core nodes**
  - **Burst lost** if…
    - **simultaneous** burst **reservations** at a core node **output** port **EXCEEDS** the available **number of wavelengths**
    - BCP and its burst are **too near in time** => no time to process BCP and switch burst



# Content

- Introduction to optical technologies: OCS-OPS-OBS
- **Which simulation platform?**
- OBS network model
  - Edge Node
  - Core Node
- The performance evaluation
  - Scenario and methodology
  - The effect of number of wavelengths on the OBS links
  - Comparison with INET
- Conclusions



# Which simulation platform?

- OMNeT++ (v3.3)
  - Do I need to explain its advantages? ;=)
- Other options:
  - OPNET:
    - (very) **expensive** annual license: give access to source of models, but **not to source of the simulator's kernel**
    - Always **fixed topology** and stored in **proprietary binary format** => difficult to use via scripts



# Which simulation platform?

- NS-2:
  - **Without separation** between kernel and simulation models
  - **It's not a simulation platform**
  - **Lacks** many tools and infrastructure components that OMNeT++ has:
    - support for hierarchical models
    - graphical user interface (**GUI**) simulation environment
    - **separation between models and experiments**
    - graphical tools for analysis
    - multiple simulation random number generator (**RNG**) streams
    - ...
  - In **Windows** it **loses** some **functionality** and you must compile and use it **through Cygwin**

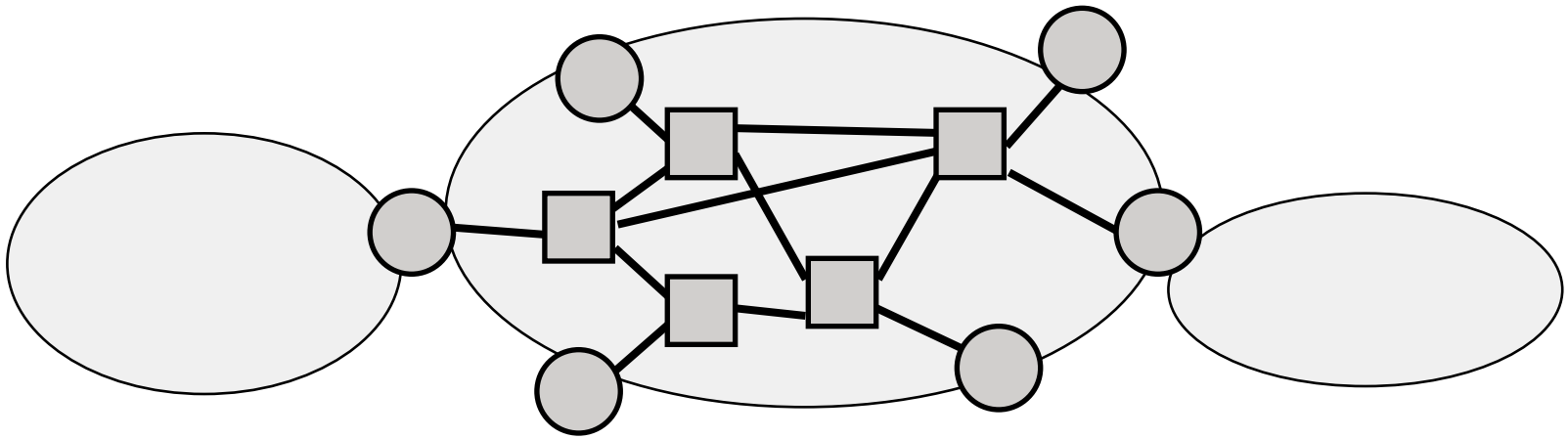
# Content

- Introduction to optical technologies: OCS-OPS-OBS
- Which simulation platform?
- **OBS network model**
  - Edge Node
  - Core Node
- The performance evaluation
  - Scenario and methodology
  - The effect of number of wavelengths on the OBS links
  - Comparison with INET
- Conclusions



# OBS network model

- There are **2 general models**:
  - Network with complete nodes:
    - All introduce (or remove) traffic
    - All have traffic in transit
  - Networks with separate nodes:
    - **Edge Nodes**: capacity to introduce (or remove) traffic in the OBS backbone
    - **Core Nodes**: only optical traffic in transit



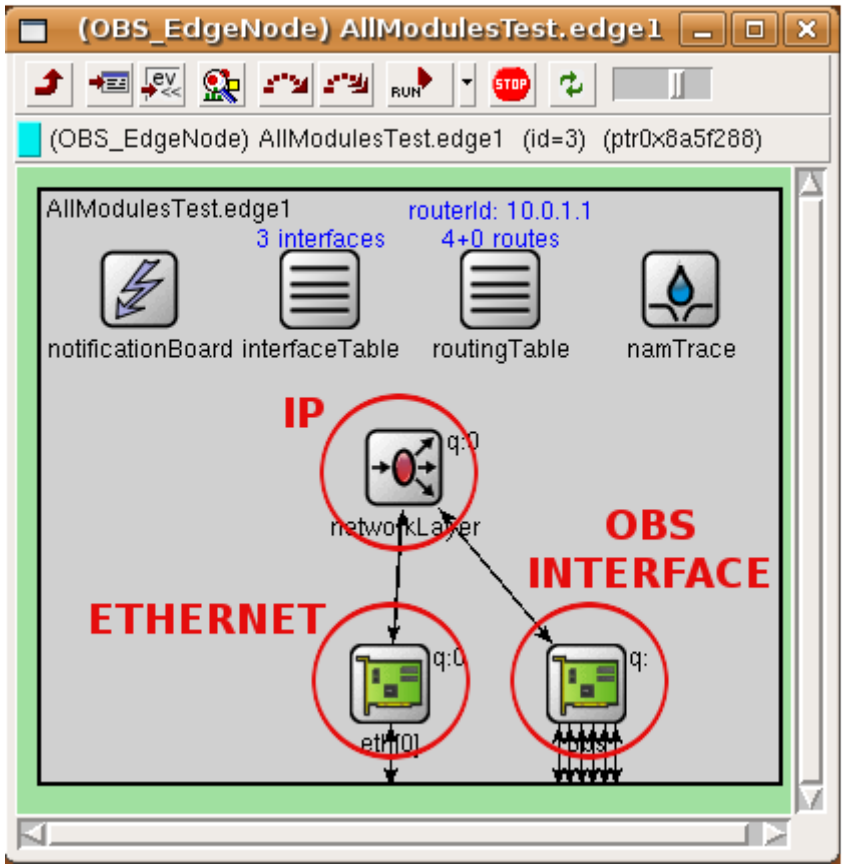
# Content

- Introduction to optical technologies: OCS-OPS-OBS
- Which simulation platform?
- OBS network model
  - **Edge Node**
  - Core Node
- The performance evaluation
  - Scenario and methodology
  - The effect of number of wavelengths on the OBS links
  - Comparison with INET
- Conclusions



# Edge Node

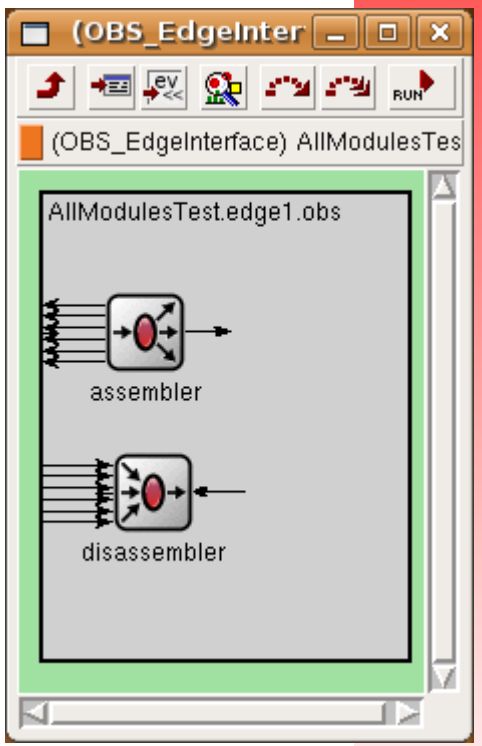
- Modelled as a **router** (INET basic router module) **with an OBS interface**



# Edge Node

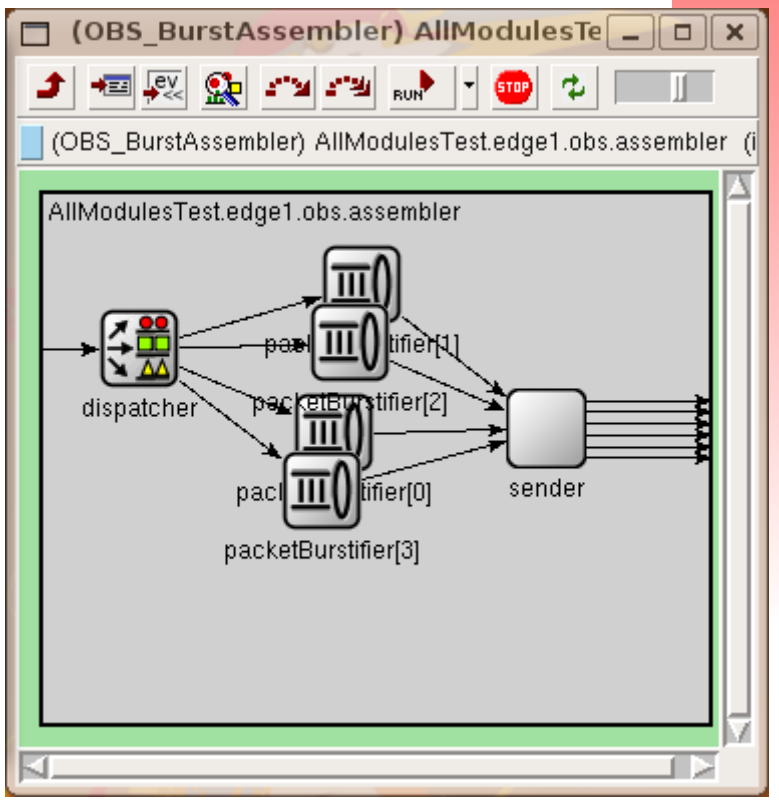
- Acts as an **ingress node** when it **introduces** traffic in the OBS network
  - **assembler module**:
    - **assemble** the incoming traffic into **bursts**
    - **schedule the transmission** of the BCP + bursts into the output channels

- Acts as an **egress node** when it **removes** traffic from the OBS network.
  - **disassembler module**:
    - performs the **inverse operation** => break down bursts into packets and forward them



# Edge Node

- **Incoming traffic aggregated (bursts)** depending on the optical destination
- This aggregation takes place in **burstifiers**
- **dispatcher** decides in which burstifiers to store the incoming traffic
- At least one burstifier per optical destination (egress)
- There could be more for differentiate traffic:
  - Ex: QoS



# Edge Node

- Implementation **supports** the most common schemes:
  - **Timer**
  - **Size**
  - Packet number thresholds
  - And the mixture of these schemes.

- **Adding a new** scheme => **only changing** the simple module **burstifier**



# Edge Node

- Optical forwarding: "label optical switching" type schema:
  - Each burst has a **label**
  - Core nodes use **label, input port and wavelength** as forwarding parameters
  - The label may change at each hop
  - **burstifier** that generates the burst **puts its label (configurable) as** the **initial label of burst**

# Edge Node

- **sender ==** OBS link level
  - It has been implemented as a **queue** in which to **store** the generated **bursts until** their **transmission**
  - Size of queue configurable for each simulation and edge node:
    - Bits
    - number of bursts
  - When a **burst cannot fit in the queue**, it is **discarded**



# Edge Node

- Used the most popular and basic OBS scheduler: **Horizon or LAUC**
  - Burst generated => **closest time when** any of the **wavelengths is free** is calculated
  - Transmission is planned for that moment
    - wavelength's horizon updated
  - Core node needs to **process BCP before** the burst arrives => BCP is sent some **offset time** before:
    - offset: maximum and minimum limits
    - Initially, BCP is planned to be sent with the maximum offset
    - If **BCP and burst are close** than minimum offset separation => **BCP and burst dropped**



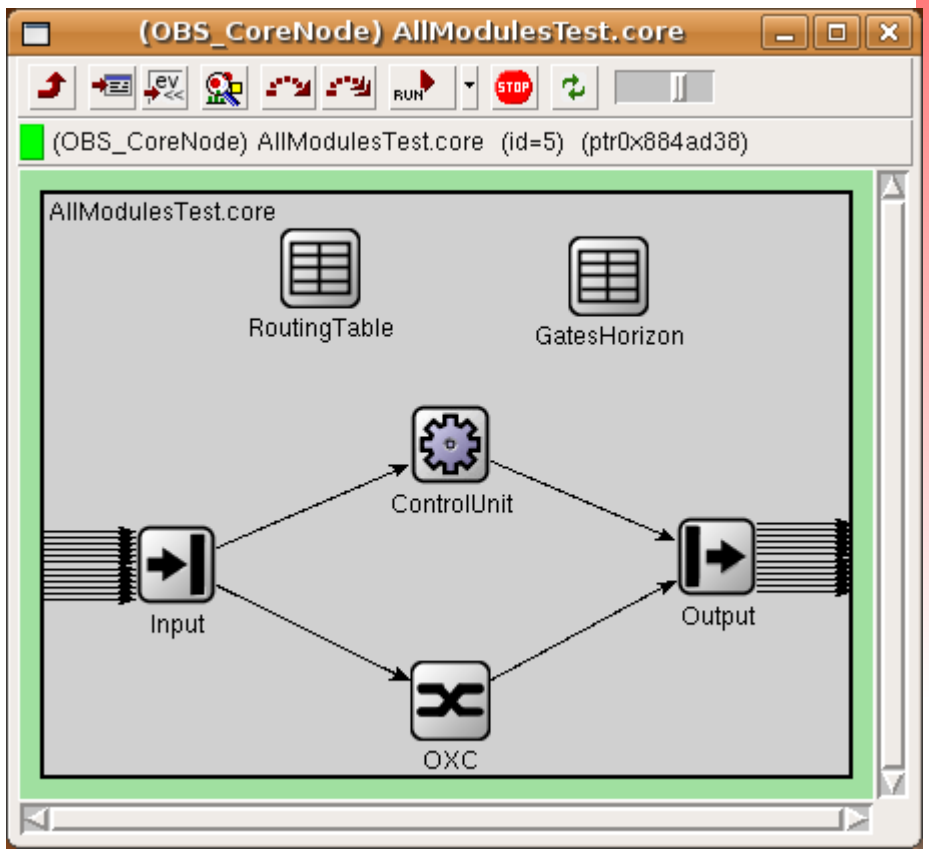
# Content

- Introduction to optical technologies: OCS-OPS-OBS
- Which simulation platform?
- OBS network model
  - Edge Node
  - **Core Node**
- The performance evaluation
  - Scenario and methodology
  - The effect of number of wavelengths on the OBS links
  - Comparison with INET
- Conclusions



# Core Node

- Responsible of:
  - **BCPs processing**
  - **bursts switching** without electro-optical conversion
  - mechanism of **contention resolution** between bursts
- Currently, it assumes that it has unlimited wavelength converters



# Core Node

- OBS signalling **typically**:
  - made **out of band**:
    - BCP uses an unique wavelength different from wavelengths for bursts
  - **and one-way initiated by the source**:
    - bursts sent without waiting for confirmation of the attempt to reserve a path
- Current implementation uses **JET scheme**
  - BCP must indicate when the burst is expected to arrive and its duration
  - Channel reservation is delayed to the estimated arrival of the burst
- Different signalling schemes have been proposed
  - Only need to change the simple module **ControlUnit**

# Core Node

- ControlUnit associates:
  - each input port, wavelength and label
  - output ports, wavelengths and labels that can use or are valid

- **Function mode:**
  - BCP arrivals => **select** the valid **wavelength** with horizon closer to and smaller than the estimated arrival time of the burst
  - **Schedule** the Optical Cross-Connect to **switch** the input wavelength with the selected output wavelength at the arrival instant and to undo once the burst crosses the switching matrix.
  - If there is **no free wavelength** => **discarded BCP and burst**

# Content

- Introduction to optical technologies: OCS-OPS-OBS
- Which simulation platform?
- OBS network model
  - Edge Node
  - Core Node
- **The performance evaluation**
  - Scenario and methodology
  - The effect of number of wavelengths on the OBS links
  - Comparison with INET
- Conclusions



# The performance evaluation

- Global performance with other OBS simulator not much useful => depends on:
  - quality of the OBS implementations
  - but also on the **different performance of simulation frameworks**
- Performance is evaluated against a similar model for OMNeT++
  - both share a common ground (OMNeT++)
  - **difference is due to the code**
- The comparing selected model: the well known **INET model**

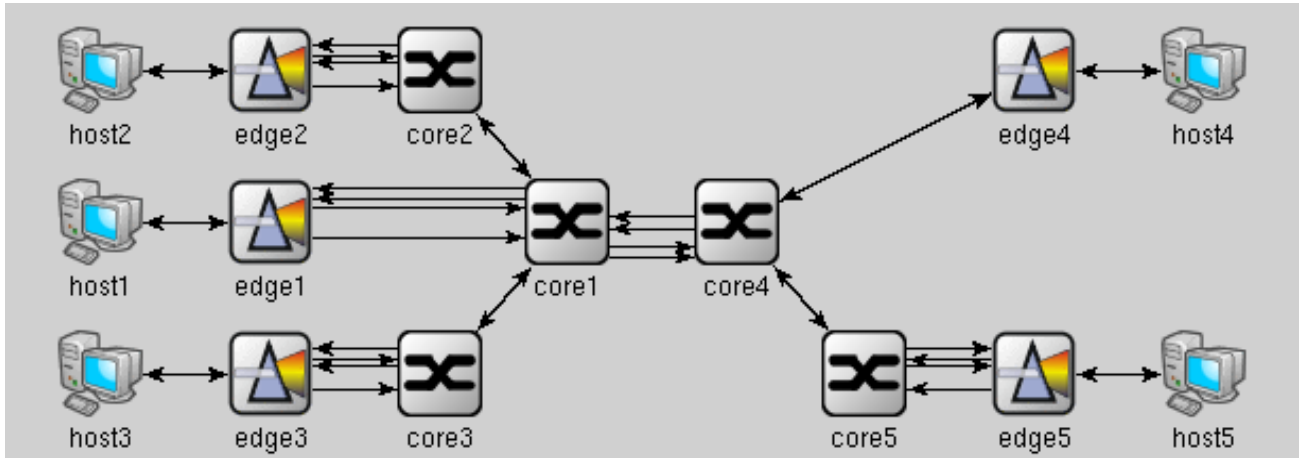
# Content

- Introduction to optical technologies: OCS-OPS-OBS
- Which simulation platform?
- OBS network model
  - Edge Node
  - Core Node
- The performance evaluation
  - **Scenario and methodology**
  - The effect of number of wavelengths on the OBS links
  - Comparison with INET
- Conclusions

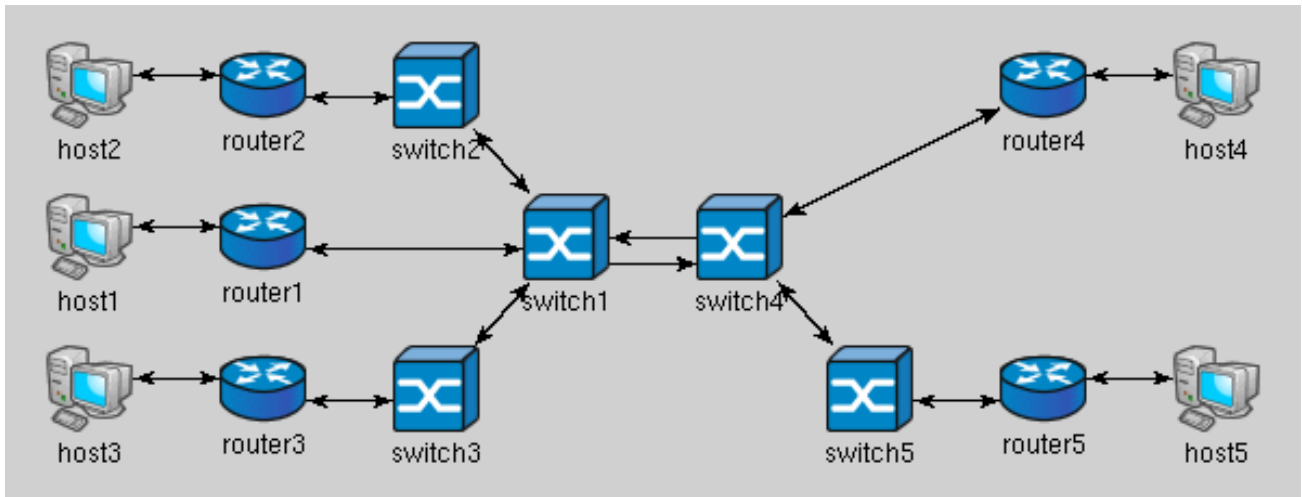


# Scenario and methodology

- OBS network scenario:




- Analogue for INET simulations with Ethernet switching technology



# Scenario and methodology

- **3 performance parameters** measured:
  - **Duration** of the simulation
  - Number of **events** of the simulation
  - **Memory** used by the simulation
    - measured recording every second the **pmap** command output
- Show if OBS model has a serious penalty
- **Simulated time: 1 minute**
  - stationary state reached in less than 1 simulated second
- Machine used: an Intel Core 2 Duo E6570 (@2.66GHz) with 3GiB of RAM and Ubuntu 8.04

# Scenario and methodology

- **OBS: timer-based burstifiers**

- Input traffic:
  - **UDP** from all to all hosts
  - **Fixed packet lengths**
  - **Poisson distribution arrivals**
    - chosen to create a preconfigured load at the central link

- Links:
  - Ethernet: 10Gbps Ethernet links
  - OBS, 2 approaches:
    - Only one data wavelength (10Gbps) per link
    - 10 data wavelengths (1Gbps) per link

# Content

- Introduction to optical technologies: OCS-OPS-OBS
- Which simulation platform?
- OBS network model
  - Edge Node
  - Core Node
- The performance evaluation
  - Scenario and methodology
  - **The effect of number of wavelengths on the OBS links**
  - Comparison with INET
- Conclusions



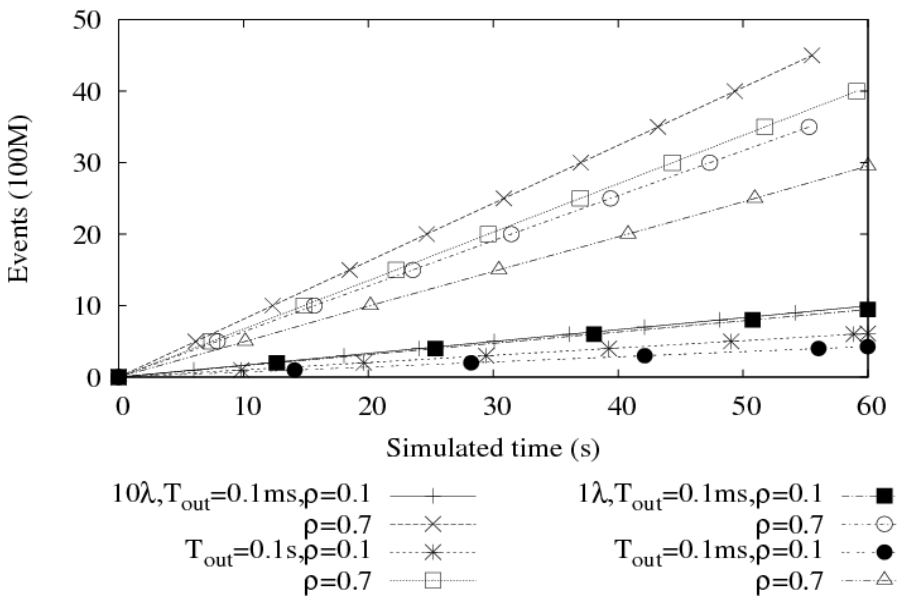
# The effect of number of wavelengths on the OBS links

- The same link capacity can be obtained using:
  - 1 wavelength of that bitrate
  - K wavelengths of bitrate/K

- Same technology... but maybe,
  - **one more expensive** (in events, time, etc.) **than other**??

# The effect of number of wavelengths on the OBS links

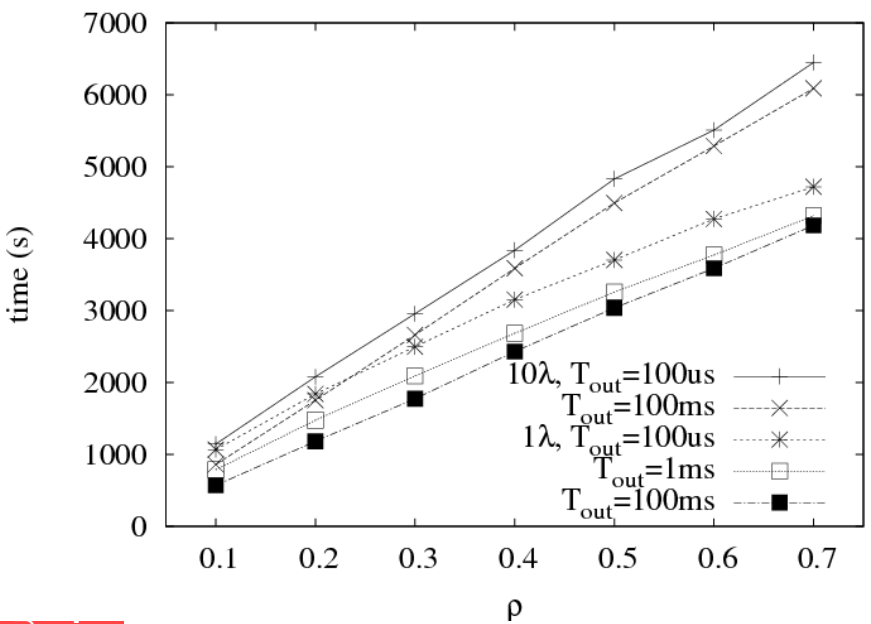
- **Events vs simulated time**, with different wavelengths, timer and load




- Events **increases linearly with load** <=> number of packets increases linearly
- **Timer increases** => number of bursts reduced (more packets per burst) => **less events**
- **Events increase with number of wavelengths**

# The effect of number of wavelengths on the OBS links

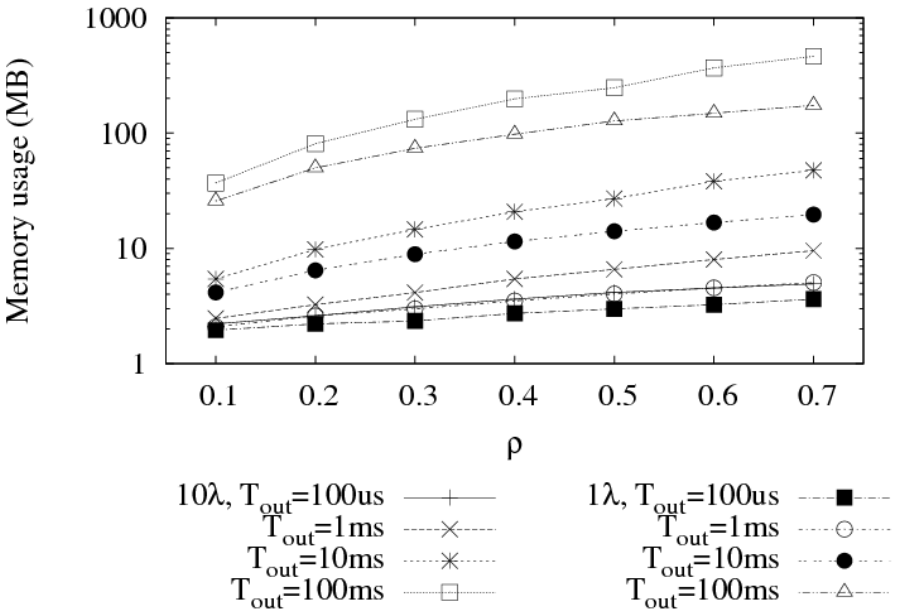
- **Processing time vs load**, with different wavelengths and timer




- **Processing time grows linearly with the load**
- Processing time also **grows with the number of wavelengths**
- More events usually implies more time and more memory

# The effect of number of wavelengths on the OBS links

- **Memory usage vs load**, with different wavelengths, timer and load




- **Memory grows with the load**: more load => more scheduled events => more memory
- **Timer increases** => number of packets inside burst and time spend inside it grows => **more memory**
- **Memory also grows with number of wavelengths**

# The effect of number of wavelengths on the OBS links

- **Conclusion**:
  - Simulation with **10 wavelengths at 1Gbps costs more** in events, simulation duration and memory used **than** a simulation with **1 wavelength at 10Gbps** and the same load and timer

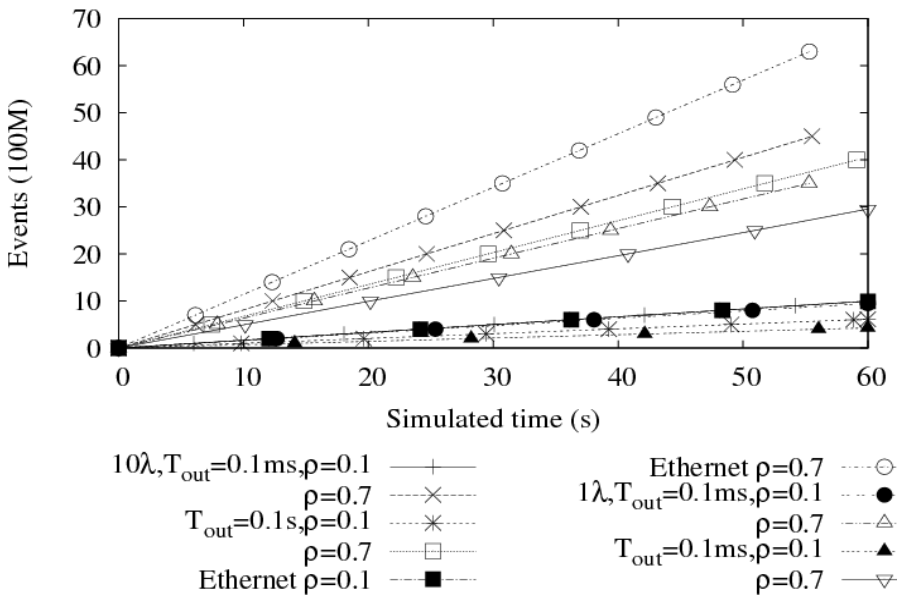
# Content

- Introduction to optical technologies: OCS-OPS-OBS
- Which simulation platform?
- OBS network model
  - Edge Node
  - Core Node
- The performance evaluation
  - Scenario and methodology
  - The effect of number of wavelengths on the OBS links
  - **Comparison with INET**
- Conclusions



# Comparison with INET

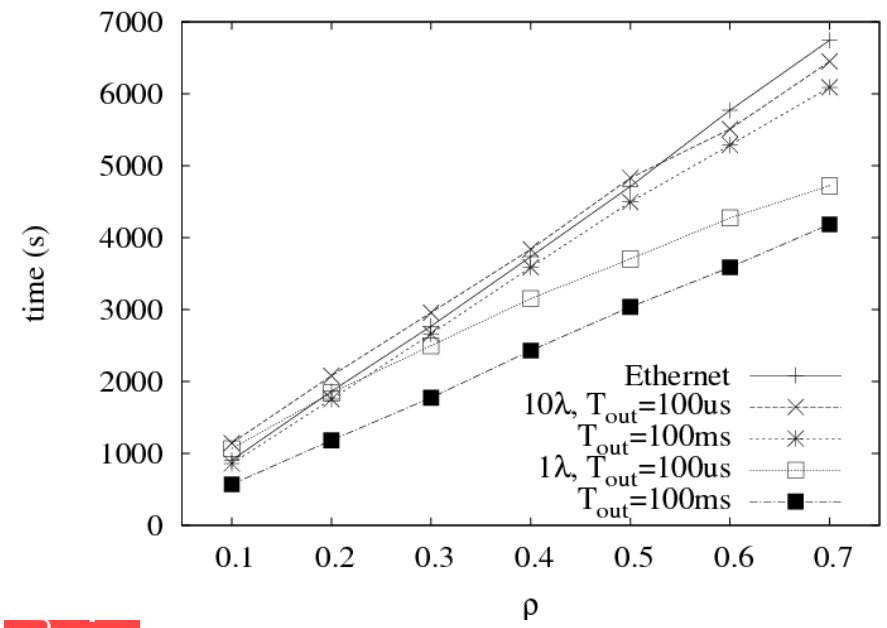
- **Events vs simulated time**, with different wavelengths, timer and load




- **INET** simulation have **at least the same** number of events
- In OBS **more packets** are inside each burst if
  - Number of wavelengths decreases
  - timer increases
  - or load increases
- => **less forwarding** work => **less events**
  - for low load this is not significative

# Comparison with INET

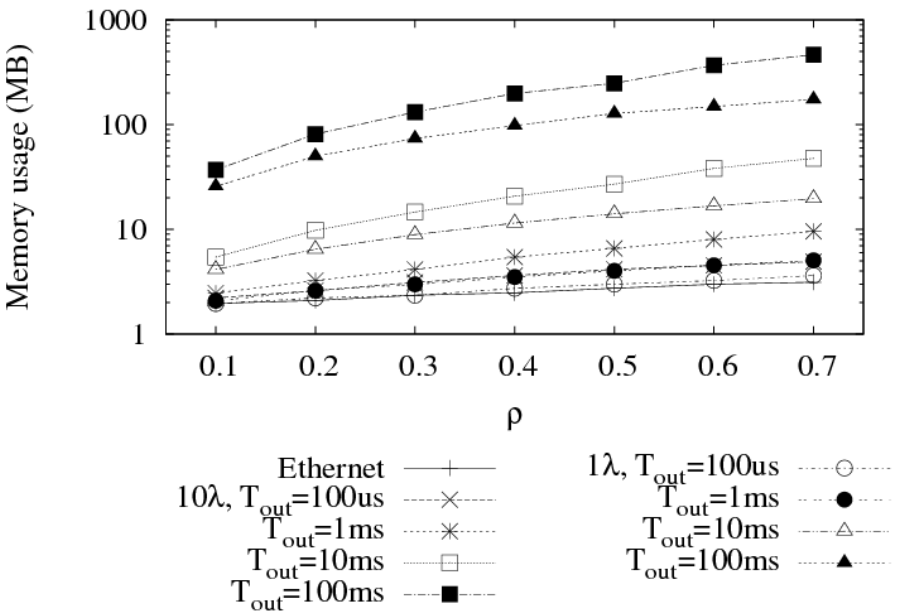
- **Processing time vs load**, with different wavelengths and timer




- For **small load INET** is as **fast** as OBS with 1 wavelength and **faster** than OBS with 10 wavelengths
- **For moderate to high load INET is always worse** => number of events to manage is always greater

# Comparison with INET

- **Memory usage vs load**, with different wavelengths, timer and load




- The **INET** model uses always **less memory** than the OBS model
- In OBS, packets travel in groups => spend more time inside simulator => increase the memory usage

# Content

- Introduction to optical technologies: OCS-OPS-OBS
- Which simulation platform?
- OBS network model
  - Edge Node
  - Core Node
- The performance evaluation
  - Scenario and methodology
  - The effect of number of wavelengths on the OBS links
  - Comparison with INET
- **Conclusions**



# Conclusions

- New OBS model for OMNeT++ was introduced
- Includes implementation of edge and core node
- Developed and implemented taking into account modularity => addition of future proposals
- The model simulates correctly the basic operations
- The performance of the OBS model was compared with the well know INET model
  - similar performance in number of events and simulation duration
  - need more memory



upna Universidad Pública de Navarra Nafarroako Unibertsitate Publikoa

# Thanks!

- More info about our OBS modules:

- **https://www.tlm.unavarra.es/investigacion/proyectos/strrong/soft/**