

**Experimentation as a research  
methodology to achieve concrete  
results: where, how, when**

Dimitri Papadimitriou (ALB)  
**ECODE (FIRE) Project**

FIA Stockholm  
November 23-24, 2009

**Context:** fundamental challenges of the Internet and its evolution

- Operational challenges
  - Manageability/diagnosability
    - Case: Network performance monitoring by combining adaptive passive and active measurements
  - Availability
    - Case: Informed path ranking based on (bandwidth x delay) predictive metrics
    - Case: Traffic (rate) informed path re-routing in case of topological failures

---

**Context:** fundamental challenges of the Internet and its evolution

- New challenges

- **Security**

- Case: cooperative/distributed detection of traffic anomalies to identify intrusions and attacks

- **Accountability**

- Case: detection of created congestion volume wrt to congestion profiles and their adaptation to maintain fair resource usage

- **Scalability and quality of routing system**

- Case: (pro-active) detection of path exploration events impacting routing system dynamics and decision on mitigation (pro-actions) reactions

# ECODE Project Outline



- **Context:** fundamental challenges of the Internet and its evolution
  - Operational challenges: manageability/ diagnosability, and availability
  - New challenges: security, accountability, and scalability & quality (routing system)

→ Derive machine learning algorithms, building blocks (architecture) and functional & system-level components

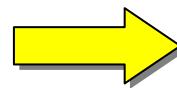
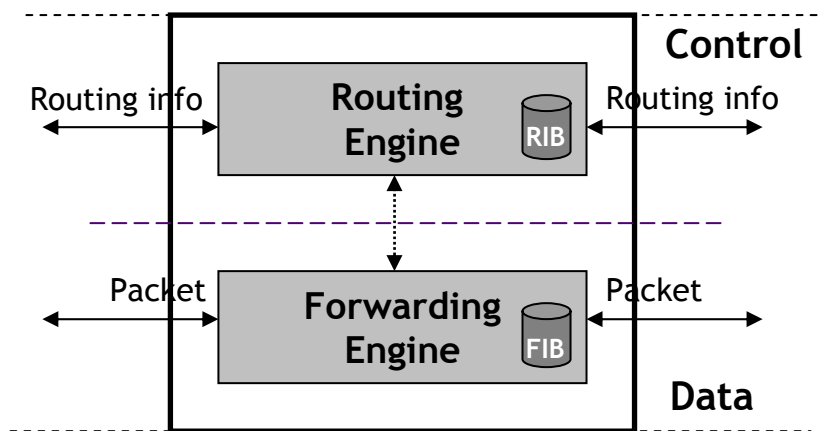
- **Objective:** cognitive routing system combining networking x machine learning technique
  - Main concept: extend IP networking equipment with a distributed machine learning engine

# ECODE Project Outline

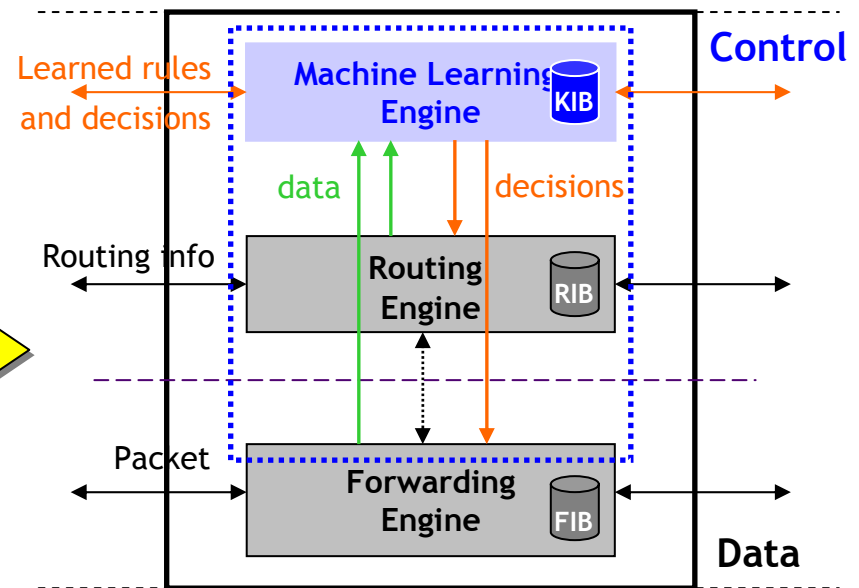
Augment control paradigm of lower-level data collection and decision making process, with **machine learning component** enabling system and network to

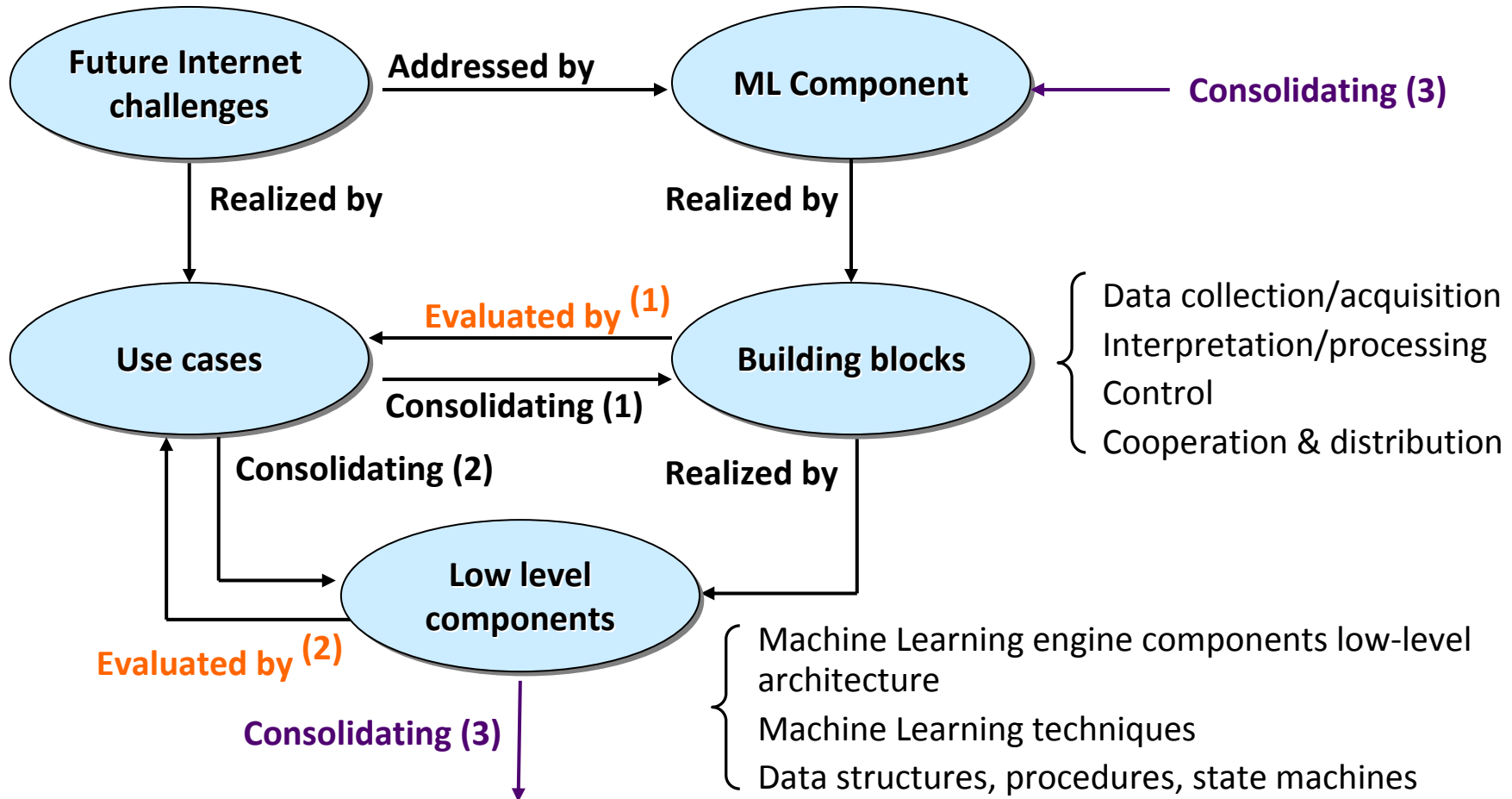
- Learn about its own behavior and environment over time
- Analyze problems, tune its operation and increase its functionality and performance

Current router design



Router with Machine Learning Engine





Experimental evaluation iLAB Virtual Wall (1) and OneLab/PlanetLab (2)

- Develop, implement and experiment semi-supervised, on-line, and distributed machine learning techniques as part of routing system
  - Improve (cost/)functional gain and (cost/)performance gain by adapting forwarding and routing system decisions
  - Implementation of functional components of the cognitive system
  - Identification of various trade-offs in distributing these components within a router
  - Design of communication protocols between distinct ML engines and how distinct ML engine cooperate in order to achieve targeted functionality

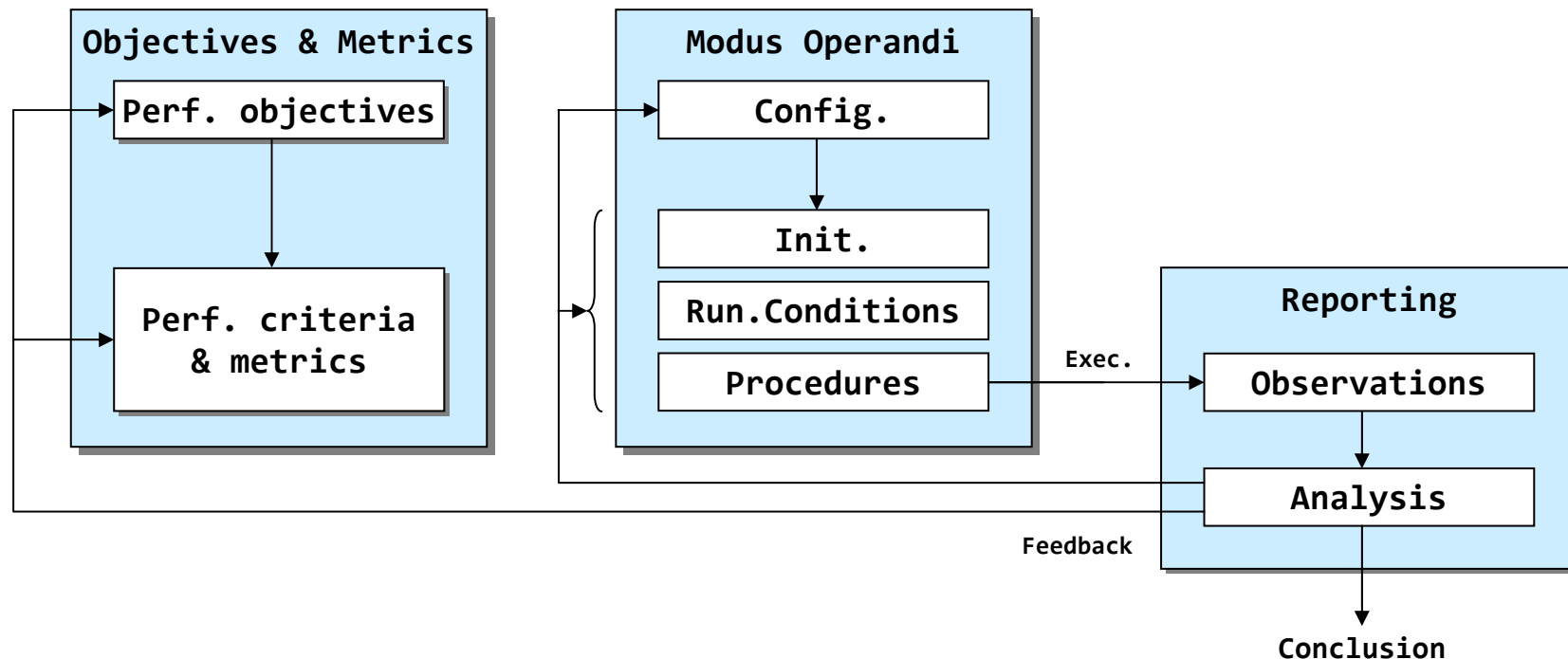
- Prototype implemented on XORP platform
  - Independence on any routing and forwarding engine
  - Open platform ( $\Rightarrow$  port experimental software over multiple experimental facilities)
  
- Iterative cycles of experimental research for progressive validation and adaptation (to cope with discovered limitations)

## Combined experimentation

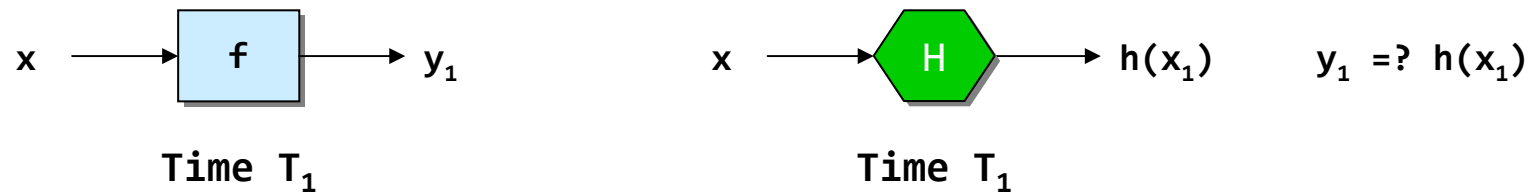
- **Phase 1:** Physical (controlled) experimental facility ([iLAB Virtual Wall](#))
- **Phase 2:** Virtual experimental facility ([PlanetLab](#))

# Experimental Methodology

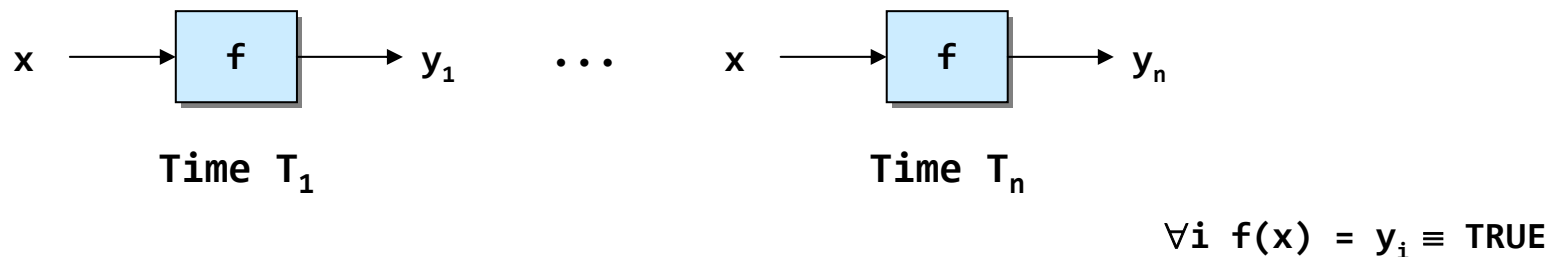
- Performance objectives, (technical and non-technical) constraints, and description of expected results
- Performance criteria and metrics
- Modus operandi: configuration, initialization, and running conditions and (iterative) procedure(s) to be executed
- Reporting including feedback on each iteration



- Experimental results (-> "function" f properties)
  - **Verifiable**: h formal model of f -> credibility

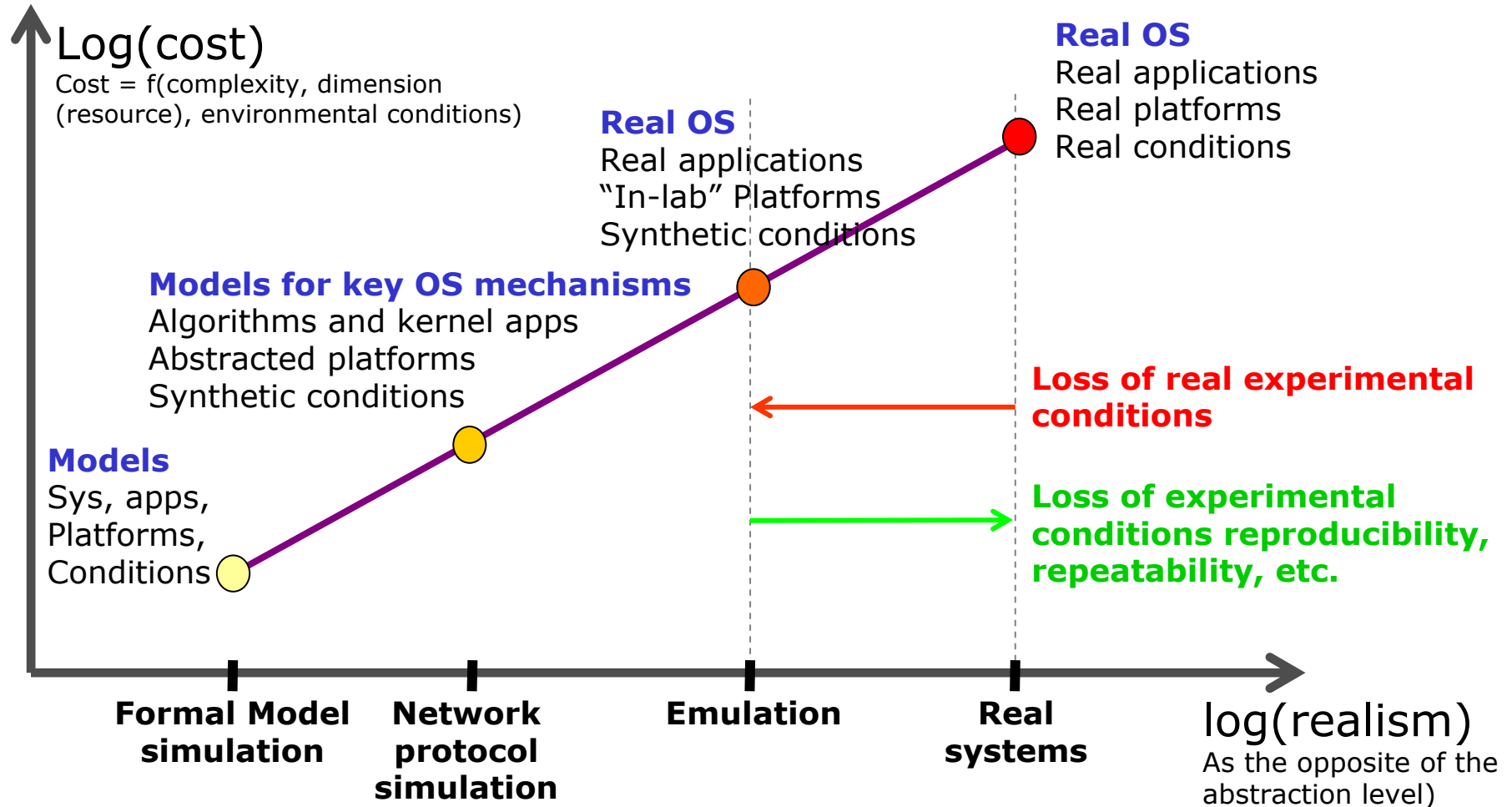


- **Reliable**: probability that system or component will perform its intended function during a specified period of time under stated conditions

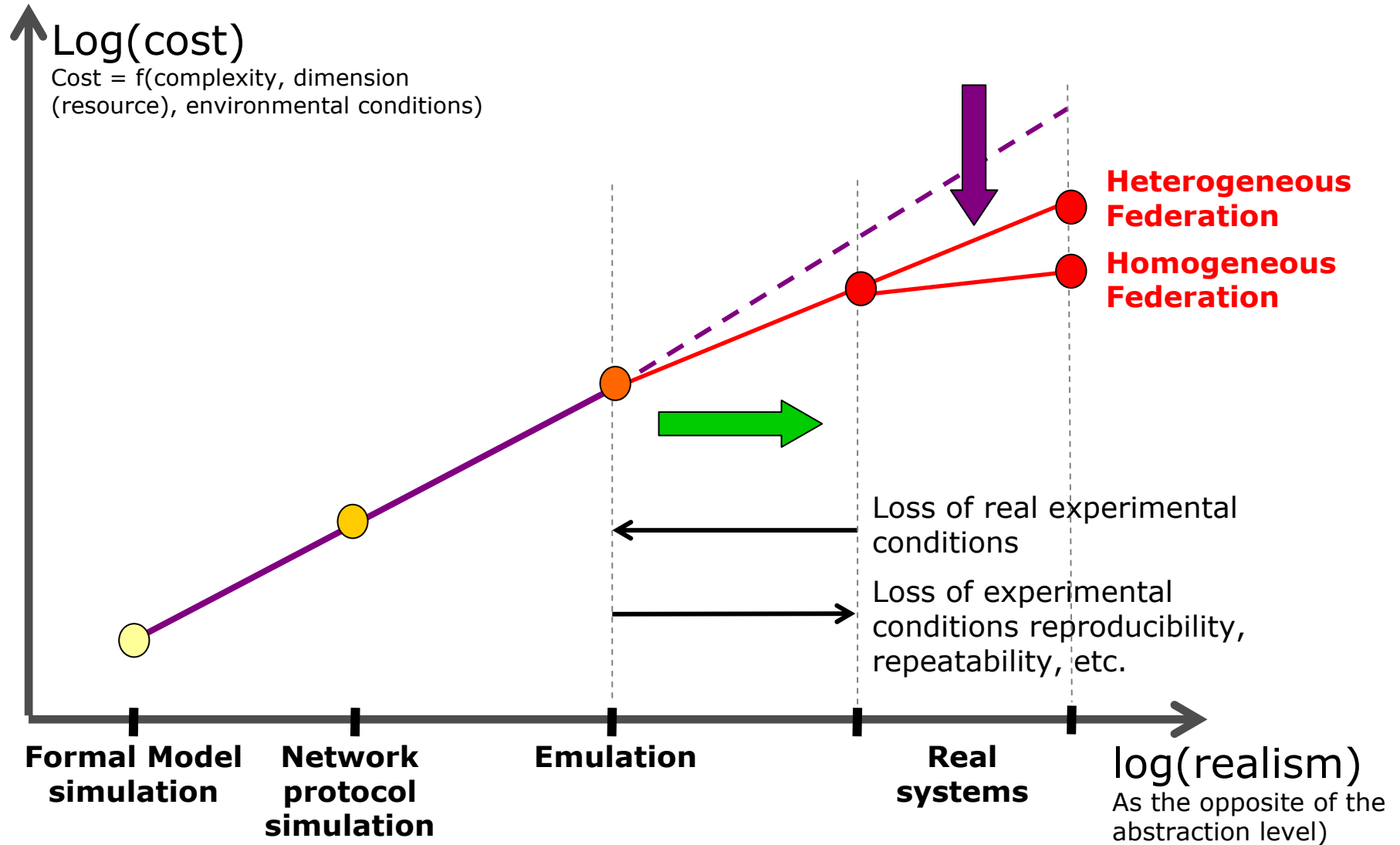




# Cost vs Level of Realism



## Abstraction and Cost vs Realism of the different experimental tools in network research



## □ Pros

- Validation of algorithms, mechanisms, ...
- Fast, and relatively simple (by tuning the complexity of models that involves links, nodes and related components)
- Reproducible and repeatable results
- Experimental environment is fully controlled
- Scalability (if appropriate tools)

## □ Cons

- Limited "realism": OS, network conditions, and traffic (models)

## □ Required at early stage of design and development process

- Model/numerical simulation: MatLab
  - Protocol component simulation: NS2
- } both used in ECODE

## □ Pros

- Controlled environment
  - Reproducible and repeatable results only if "conditions" and "executions" can be controlled
- Realism can be improved compared to simulation (in particular for time-controlled executions of protocol components on real OS)

## □ Cons

- Experiments are more complex and time consuming to configure and execute
- Scalable only if platform comprises “sufficient number” of machines (representative)
- Synthetic network conditions (models)
- Background traffic must be realistic
  - **Major issue:** Real traffic traces are required  
note: even when available "spatial distribution" of traffic remains problematic

# To reach these objectives ...

---

- Develop tools and languages to
  - Describe and automatically setup experiments in several experimental facilities
    - Define standard for experiment description and control interface and wrap tools within this API (integration and unification)
      - Input (x), Function (f), Output (y)
      - Resources (on which f executes)
    - Provide common programming interface to describe every aspect of a networking experiment
  - Collect monitoring data (such as traces) to allow repeatable, reproducible, verifiable, reliable, and ultimately comparable results

# To reach these objectives ...

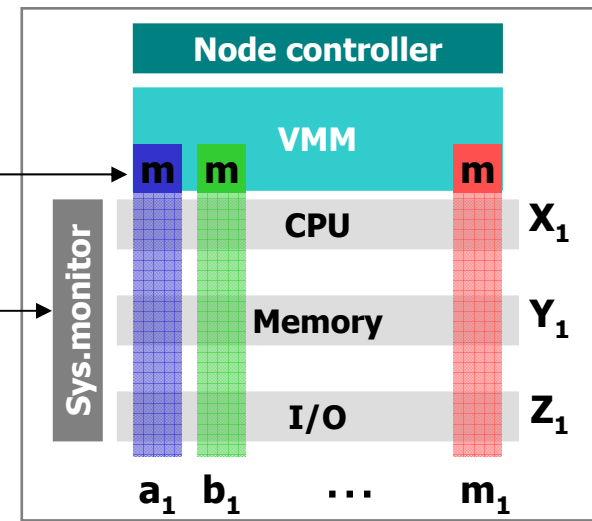
- Specify distributed performance monitoring system: robust experiment monitoring and management capability

System resources

- CPU
- Memory
- I/O

Per VM monitor

System monitor (hardware)



- Specify performance analysis methodology together with necessary tools to perform data analysis and mining tasks on data coming from various monitoring points (from single or multiple testbeds)

# To reach these objectives ...

- Develop tools to analyze sensitivity of performance measures to changes in "experimental model" parameters
  
- > **Sensitivity analysis**: identify how results of an experimental model are responsive to changes in its (input or structural) parameters
  - Fundamental tool for achieving confidence in experimentation and making its results **credible**
  - Sensitivity analysis
    - Quantifies dependence of system behavior on parameters that affect modeled process and in particular its dynamics
    - Determine how sensitive a model is to changes in the numerical value of the model input parameters and changes in the model structural parameters