TEMIC: a New Cooperative Platform for Industrial Tele-Maintenance

Mohamed Brahma[1], Martine Chaudier[2], Eric Garcia[3], Jean-Patrick Gelas[2], Hervé Guyennet[3], Fabien Hantz[3], Laurent Lefèvre[2], Pascal Lorenz[1], Henri Tobiet[4]

1. Université de Haute Alsace IUT- Colmar - France
2. INRIA RESO / LIP (UMR CNRS, ENS, INRIA, UCB), Ecole Normale Supérieure LYON - France
3. LIFC Laboratoire d'Informatique de l'Université de Franche-Comté Besançon - France
4. SWI – SA Mulhouse- France

Laurent.lefevre@inria.fr

DFMA 2006 – Penang, Malaysia – May 16, 2006

Motivations

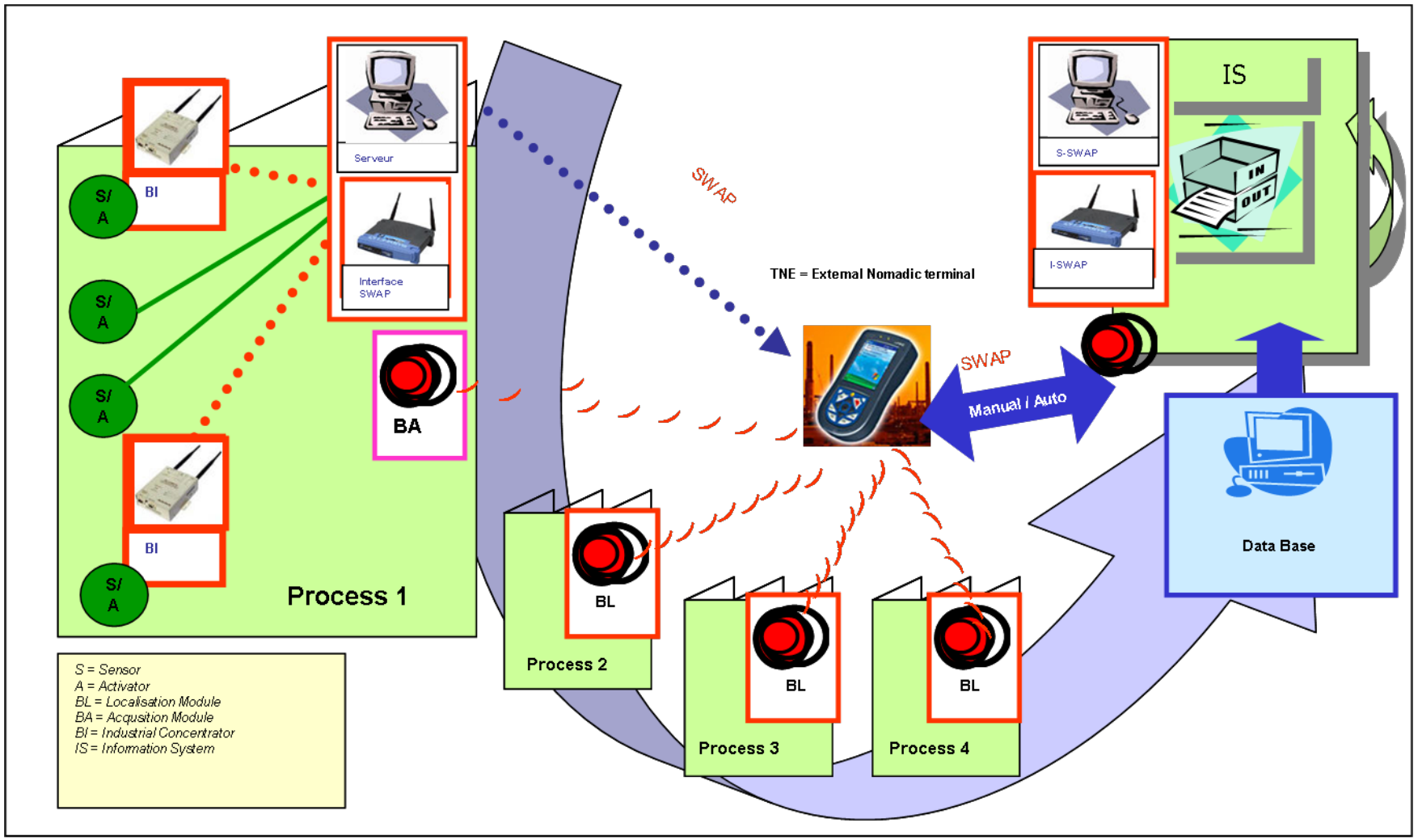
- Proposing new solutions for industrial maintenance
  - A set of sensors (even multimedia) to supervise an industrial process
  - Experts and maintenance technical people with mobile equipments (PDA, tablets, cellulars)

- TEMIC project : funded by RNRT

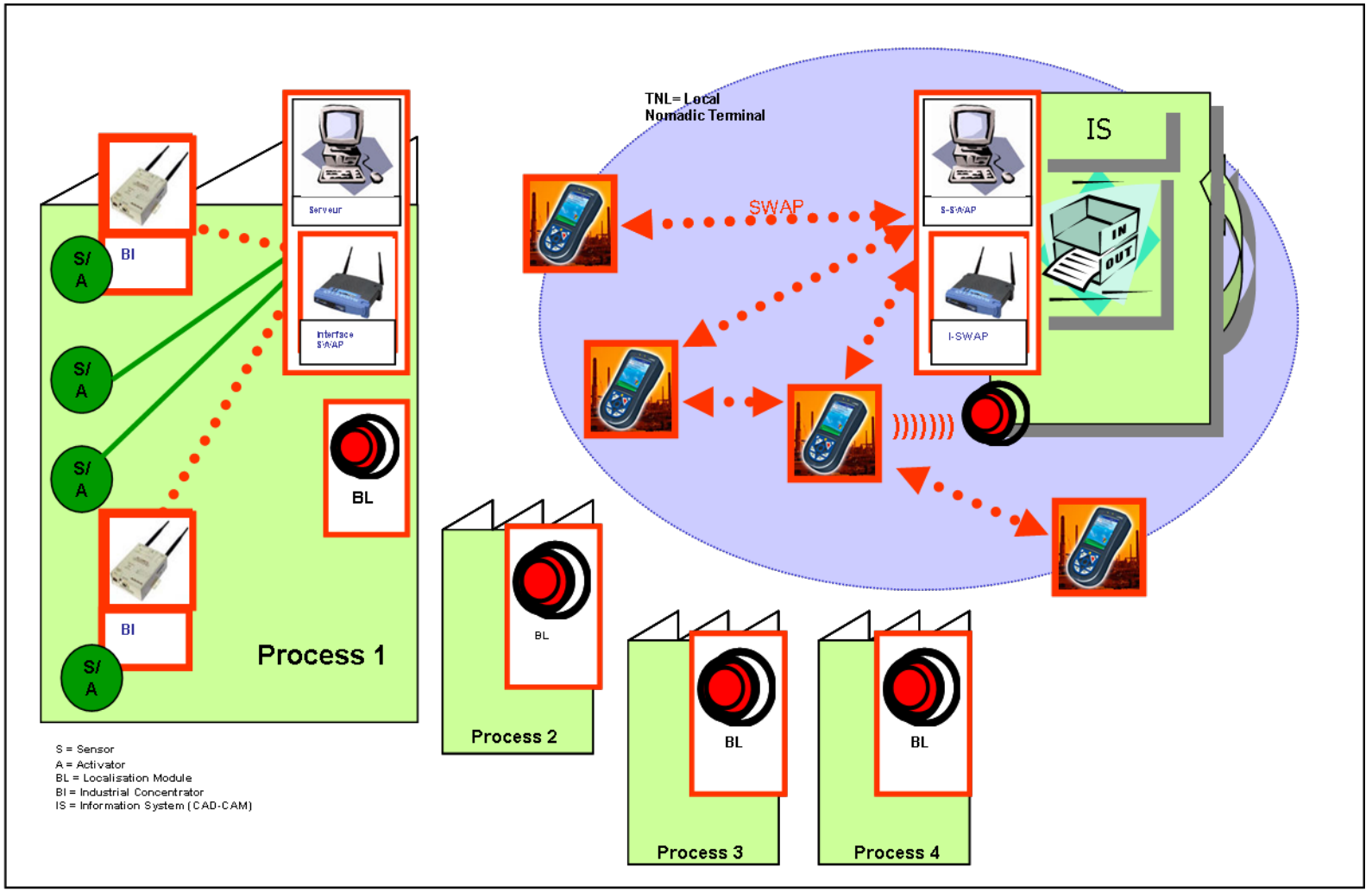
TEMIC architecture

- Three-level architecture:
  - Collection and Surveillance/Supervision,
    - allows data collection from sensors and different types of industrial interfaces,
    - performs wireless data acquisition via Wi-Fi adhoc networks,
    - is dedicated for real-time process supervision and control.
  - Intervention and Reporting,
    - allows maintenance management via mobile terminals (e.g. PDA),
    - performs wireless information reporting on maintenance activities,
    - is dedicated for real-time system maintenance (maintenance management software).
  - Diagnosis and Collaboration
    - allows collaborative actions via extended networks and communication facilities (e.g. videoconferencing via the Internet),
    - performs decisional diagnostics and forwards corrective actions to recover the process.

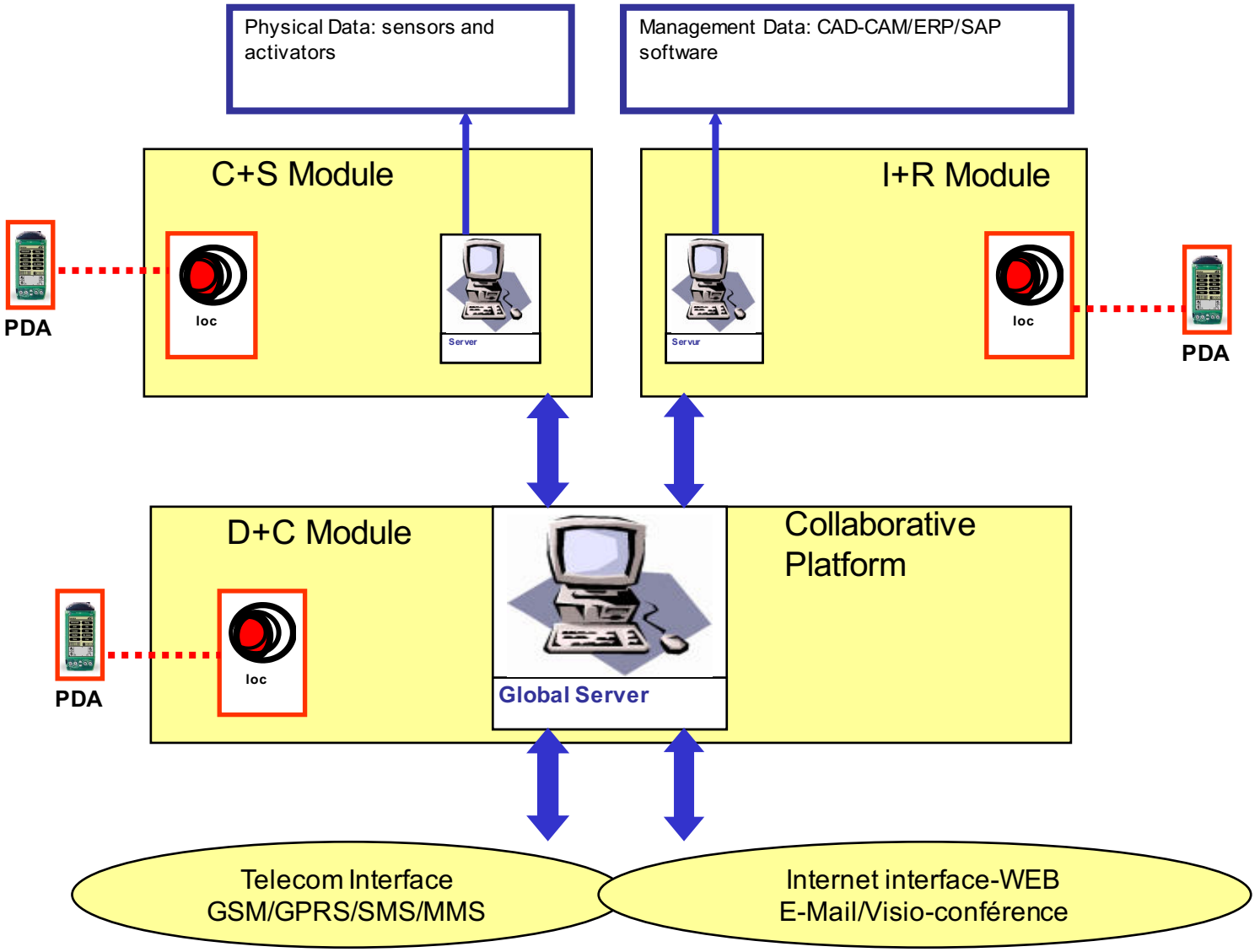
Collection / Supervision



Intervention and reporting



Whole TEMIC architecture



Let’s now focus on a small part of TEMIC project : Collection and monitoring through autonomic facilities

Adaptation services in the network : Scenario requirements

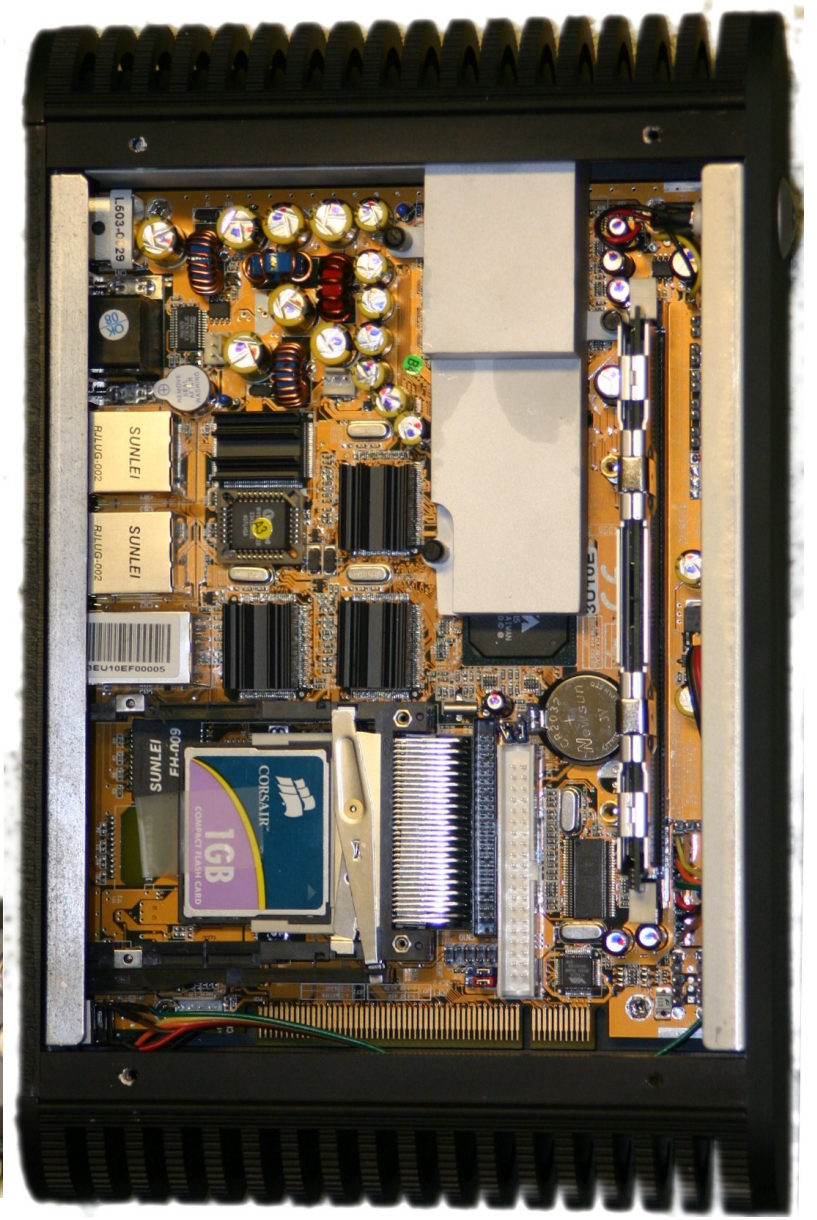
- Easily and efficiently deployable hardware in industrial context.
- Easily removable at the end of the maintenance and monitoring contract.
- Devices must fit industrial requirements:
  - reliability
  - fault-tolerance
- Devices must be *autonomic*!
  - auto-configurable
  - re-programmable

Our approach

- Designing an Industrial Autonomic Network Node (IAN^2):
  - Using a reliable and embedded hardware
  - Running on a low resource consumption node OS
  - Proposing an adapted EE
  - Designing a set of services
  - Evaluating solution in controled and industrial scenario

Hardware platform

- A transportable solution.
- Reduced risk of failure:
  - fanless
  - no mechanical hard disk drive
- VIA C3 1GHz, 256MB RAM, 3xNIC Gbit Ethernet, 1GB Compact Flash,...



Software Execution Environment:

Node Operating System

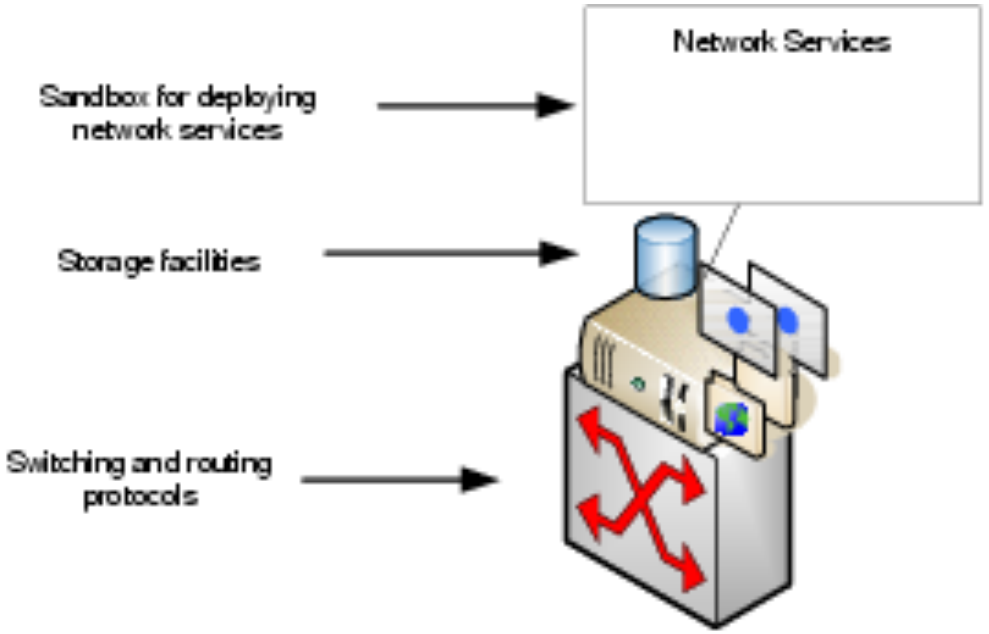
- Indutrial Autonomic Network Node (IAN^2) runs over Btux (bearstech.com)
- Btux is based on a GNU/Linux OS (kernel 2.6.12)
  - rebuilt from scratch
  - small memory footprint
  - reduced command set available
  - remotely upgradable

Software Execution Environment:

IAN^2 Software Architecture

Our Industrial Autonomic Nework Node architecture supports:

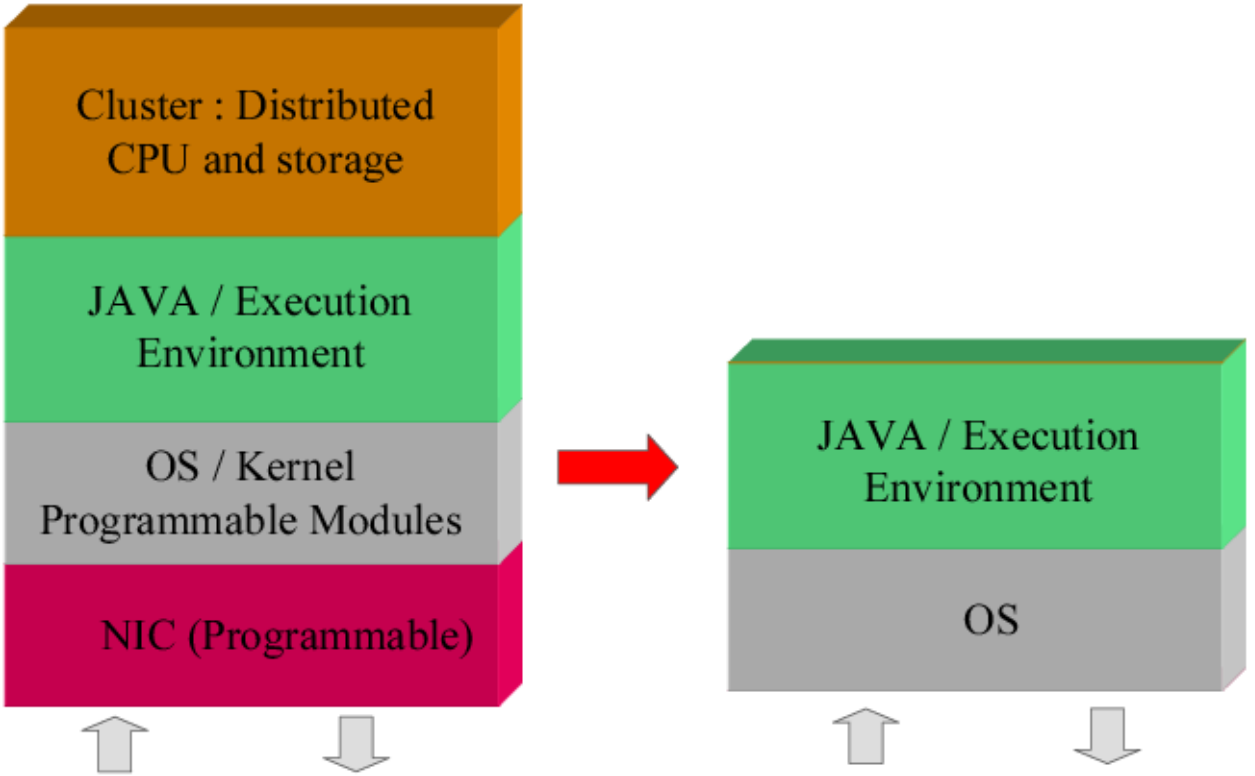
- wired and wireless connections,
- CPU facility,
- storage capabilities.



Software Execution Environment:

Execution Environment

- The EE is based on the *Tamanoir (INRIA)* software suite, a high performance execution environment for active networks.
- Tamanoir: Too complex for industrial purpose.
- ***Tamanoir*^{embedded}:**
  - reduced code complexity,
  - removed unused class and methods,
  - simplify service design.

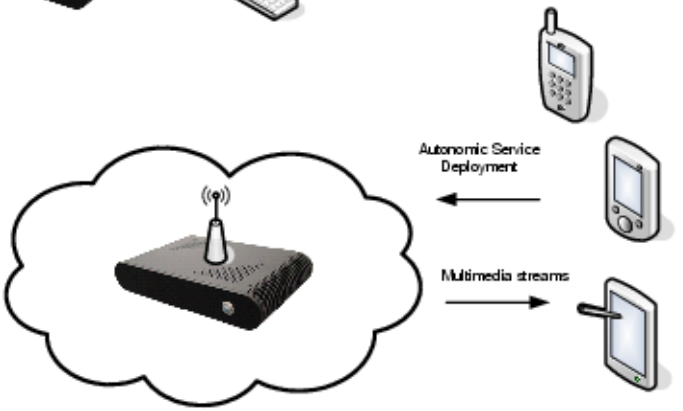


Software Execution Environment:

Autonomic Service Deployment

- ***Tamanoir^{embedded} is written in Java and suitable for heterogeneous services.***
- ***Provides various methods for dynamic service deployment/update:***
  - ***from a service repository to a Tamanoir Active Node (TAN),***
  - ***from the previous TAN crossed by the active data stream,***

  - ***from mobile equipments.***



Experimental Evaluation:

Network Performances

- Based on *iperf* (bandwidth, jitter, loss) on two topologies.

- IAN^2 failed to obtain a full Gbit bandwidth due to the limited embedded CPU and chipset.

Configuration	Throughput	cpu send	cpu recv	cpu gateway
back-2-back	488 Mbps	90%	95%	N/A
gateway (1 stream)	195 Mbps	29%	28%	50%
gateway (8 streams)	278 Mbps	99%	65%	70%

Experimental Evaluation:

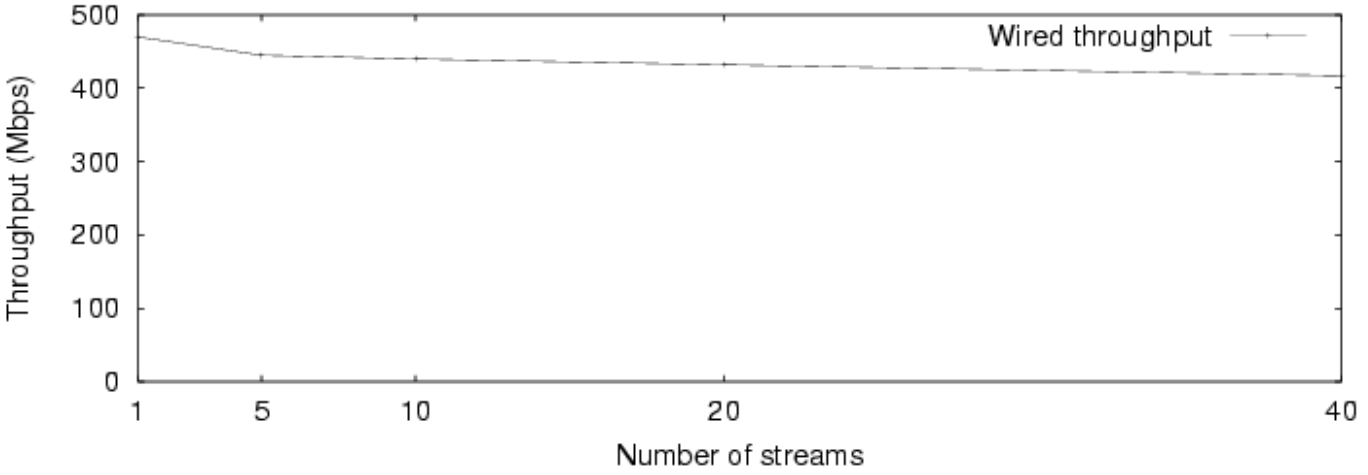
Network Performances

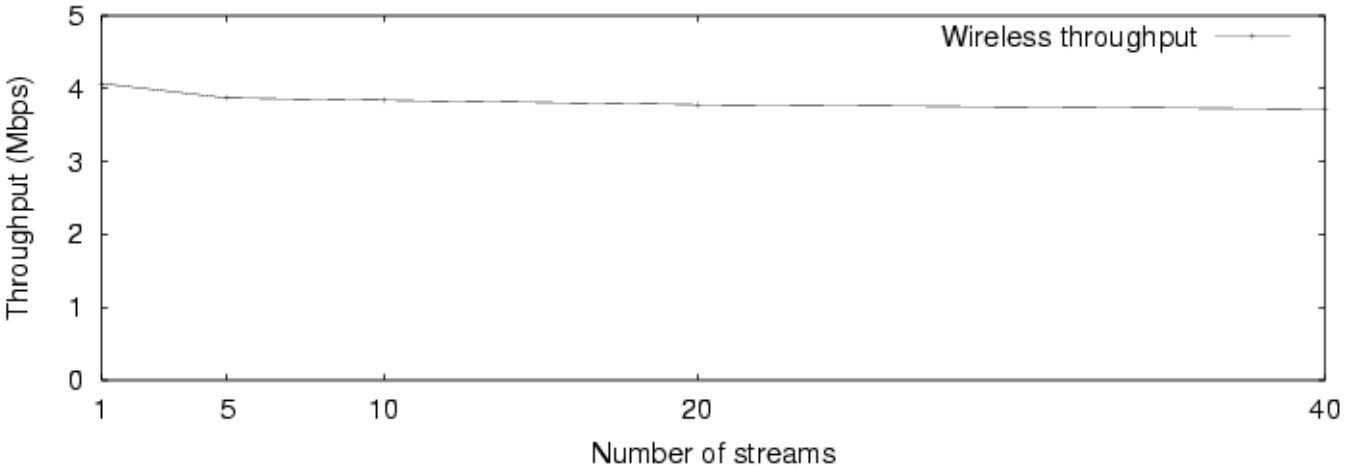
- GigaEthernet:

  480 Mbps

- Wireless (802.11b):

  4 Mbps





Experimental Evaluation:

Autonomic Performances

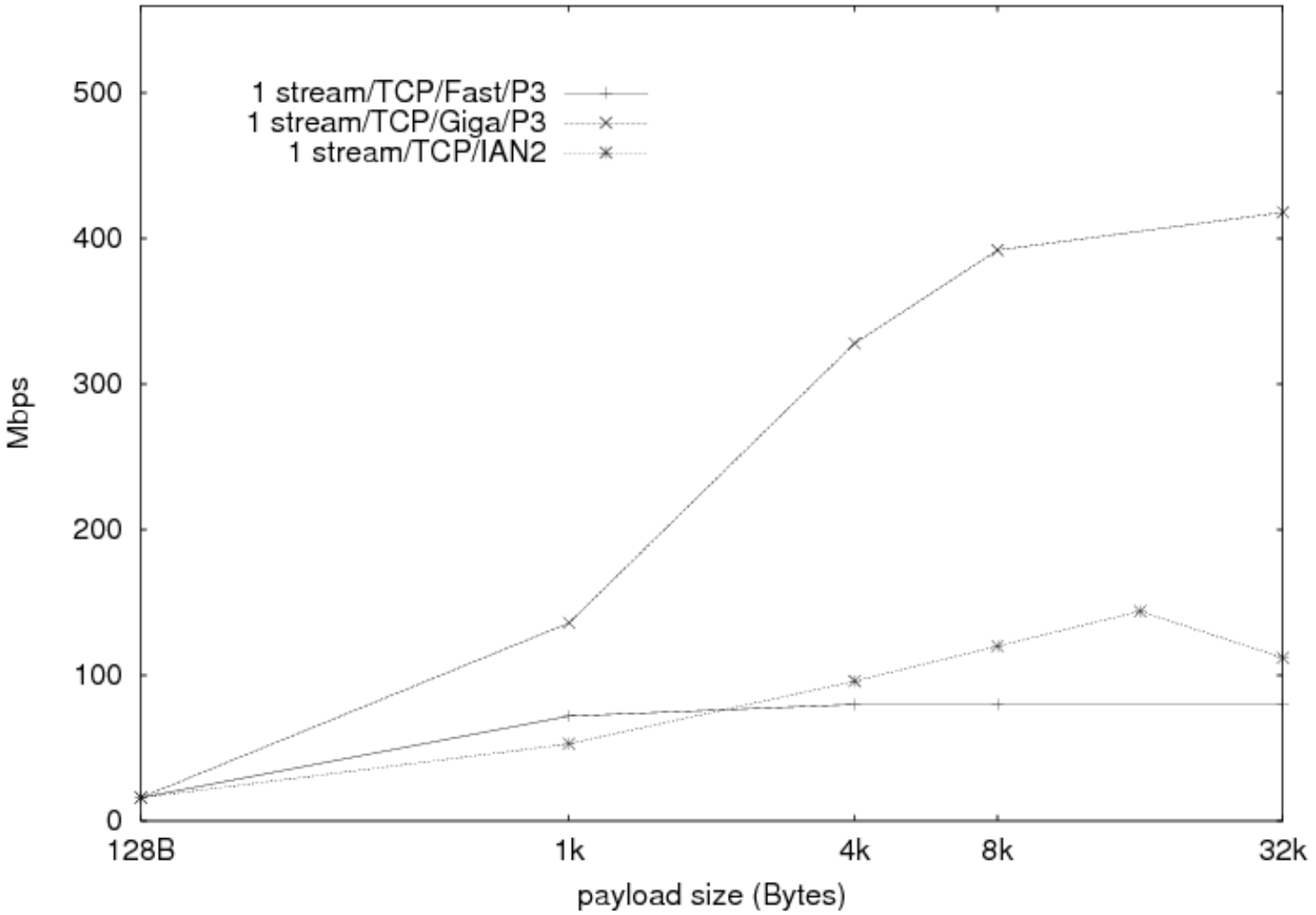
- We ran two different active services:
  - A lightweight service (MarkS)
  - A heavyweight service (GzipS)
- EE and services run in a SUN JVM 1.4.2

	4kB	16kB	32kB	56kB
MarkS	96	144	112	80
GzipS	9.8	14.5	15.9	16.6

(Throughput in Mbps)

Experimental Evaluation:

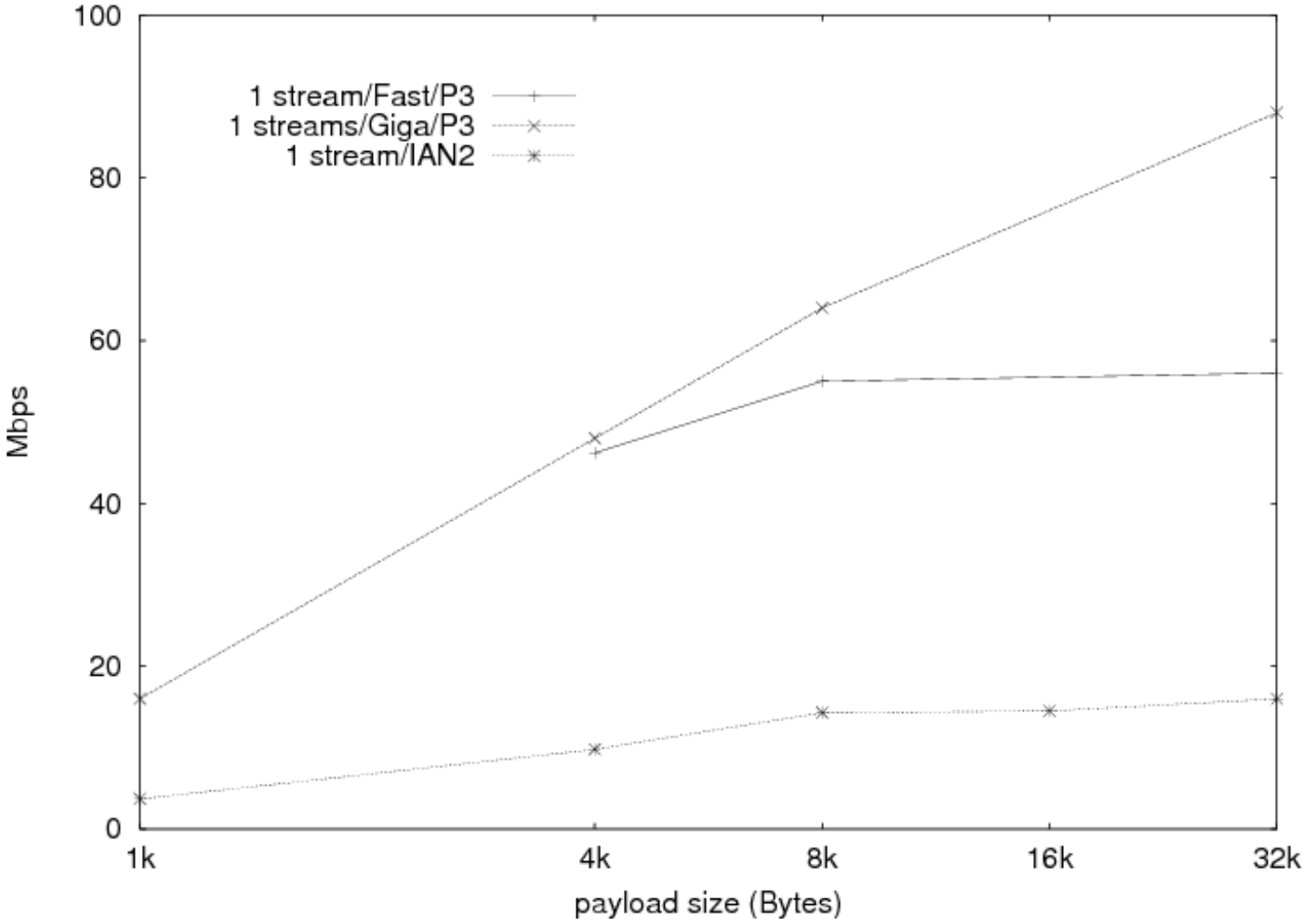
Autonomic Performances



- Performance comparison with standard system over lightweight service.

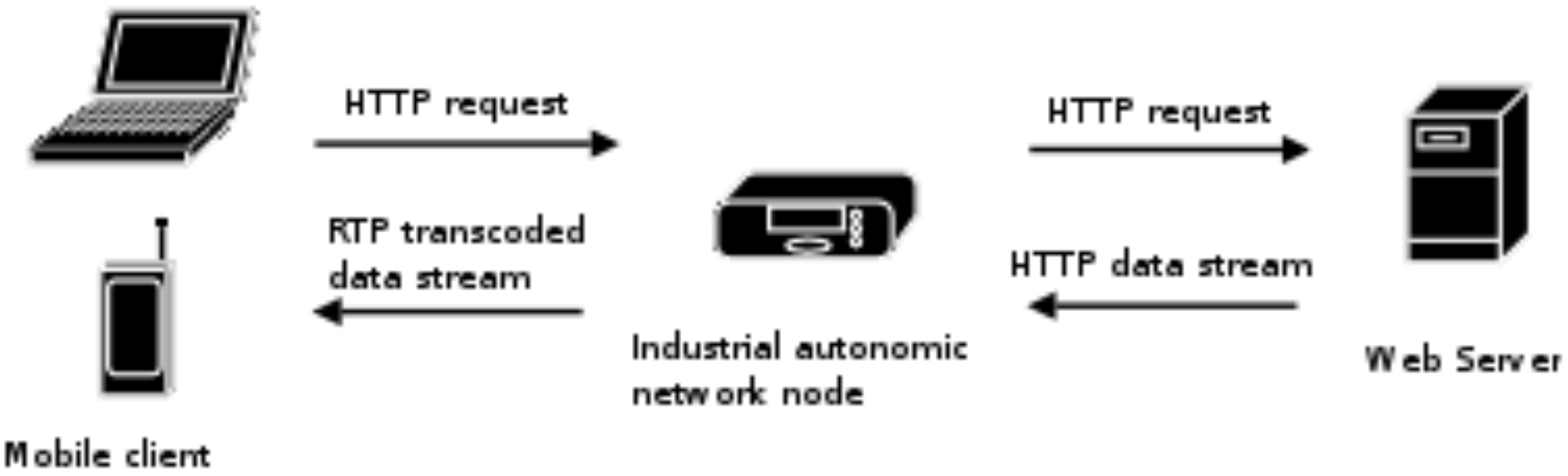
Experimental Evaluation:

Autonomic Performances



- Performance comparison with standard system over heavyweight service.

Scenario : adaptation with videostreamS



Performances within Multimedia context application

- Transmit and adapt a video stream: a real evaluation of our industrial autonomic node.
- Without adaptation step, CPU use is negligible.

Format / Size	Usr CPU load
MJPEG/720x480	< 1 %
H263/352x288	98,7 %
H263/176x144	99,3 %
H263/128x96	99 %

- Then, CPU load is totally due to the processing.

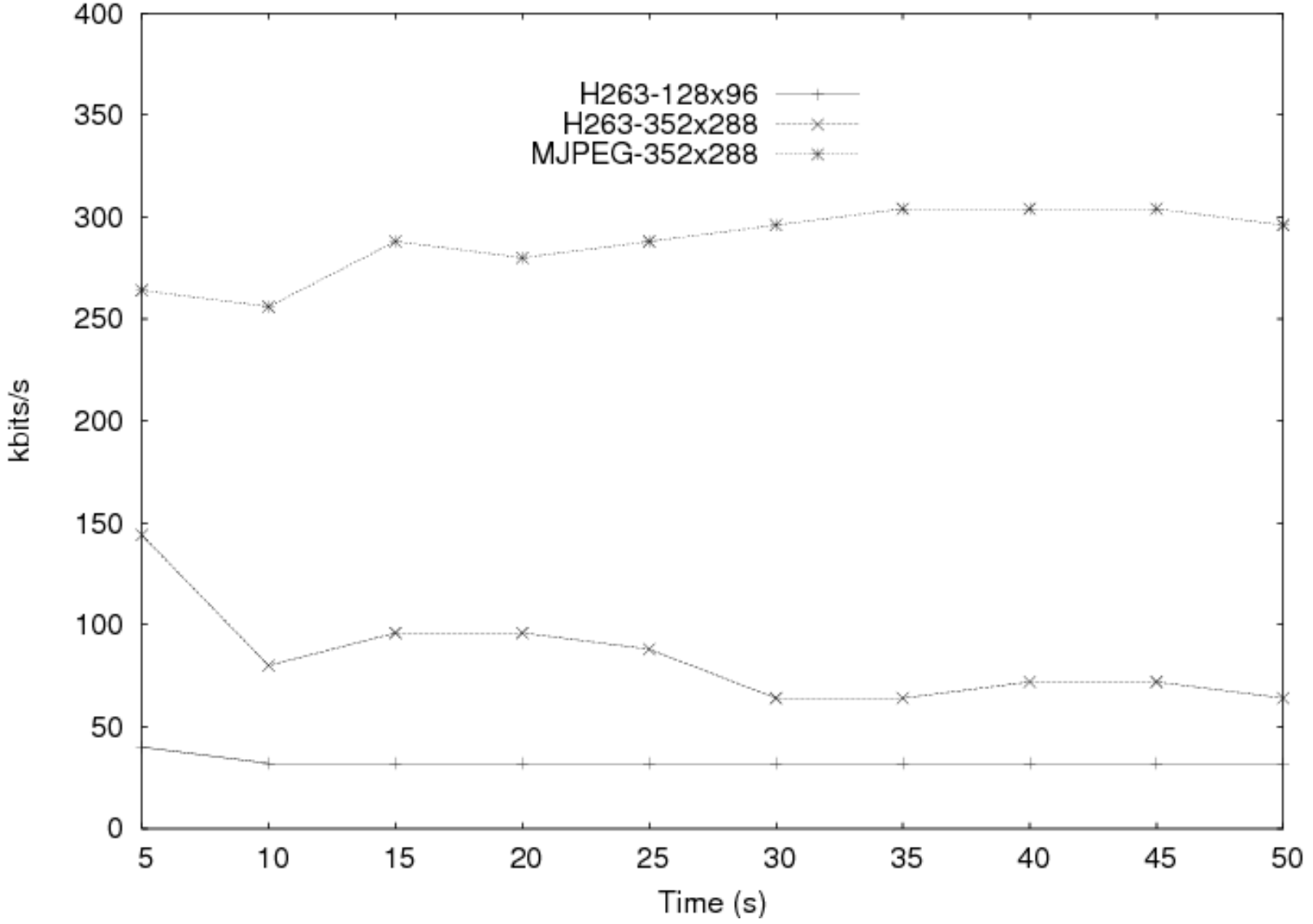
Performances within Multimedia context application

- We measure the ouput data rate on a active node using a wireless network when transmiting an adapted video file to a PDA.

Output Format/Resolution	Entry File/Output File	Transmitting time	PDA loading time
MJPEG/720x480	14794 KB / 14794 KB	4 min 50 sec	5 min 10 sec
H263/352x288	14794 KB / 1448 KB	22 sec	2 min 55 sec
H263/176x144	14794 KB / 365 KB	8,5 sec	1 min 30 sec
H263/128x96	14794 KB / 179 KB	3,8 sec	1 min 18 sec

- Even with a limited CPU, the IAN^2 provides efficient adapatation which reduces the amount of transported data and globally improves performances of the application.

Performances within Multimedia context application



Conclusions

- Description of the IAN^2 prototype of industrial autonomic network node
  - hardware choice
  - software solution
- Evaluation of performances
  - processing power
  - networking
  - Execution Environment
- Results are far from a current desktop performances, however for « low » bandwidth network (Fast Ethernet, xDSL or Wireless networks, sensors networks), IAN^2 can perfectly support a large class of reliable autonomic services.

TEMIC future works

- Current deployment on industrial sites
- Switching from academic (experimental) project to an industrial project is a real challenge.
- Next step concerns the development of new autonomic services.

Questions?

Laurent.lefevre@inria.fr