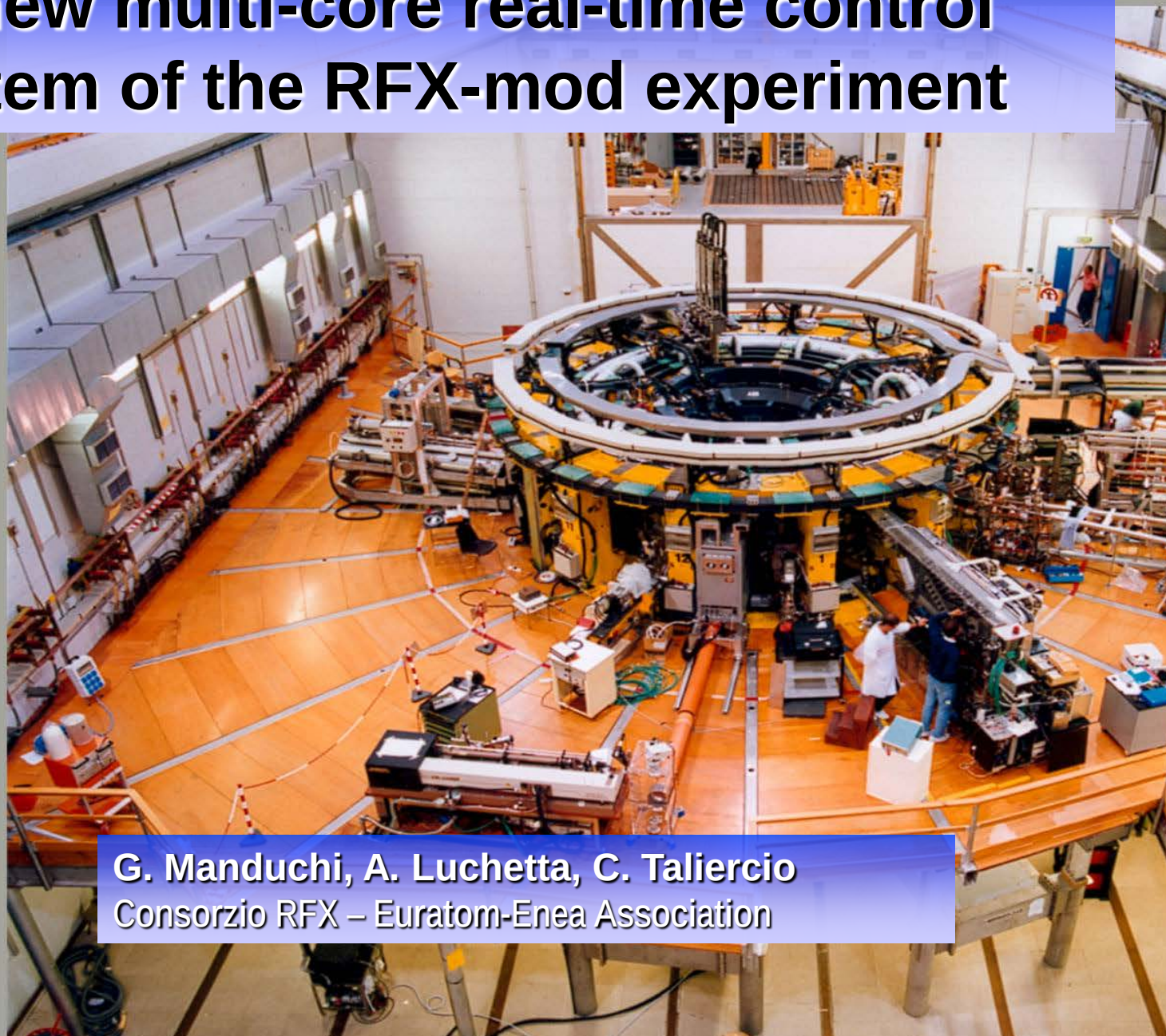
The new multi-core real-time control system of the RFX-mod experiment

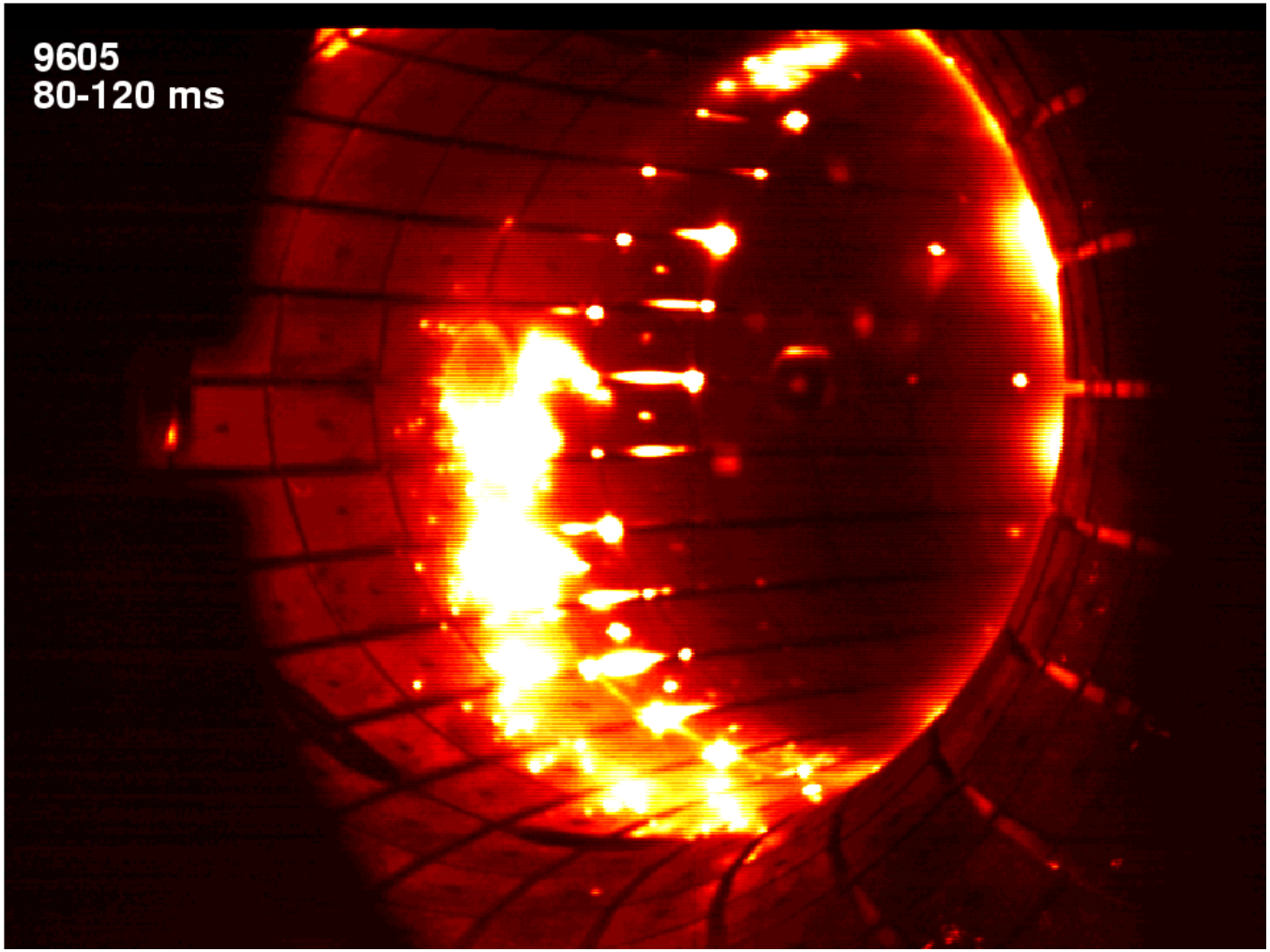


G. Manduchi, A. Luchetta, C. Taliercio
Consorzio RFX – Euratom-Enea Association

CONSORZIO RFX
Ricerca Formazione Innovazione

Why Plasma control?

CONSORZIO RFX
Ricerca Formazione Innovazione



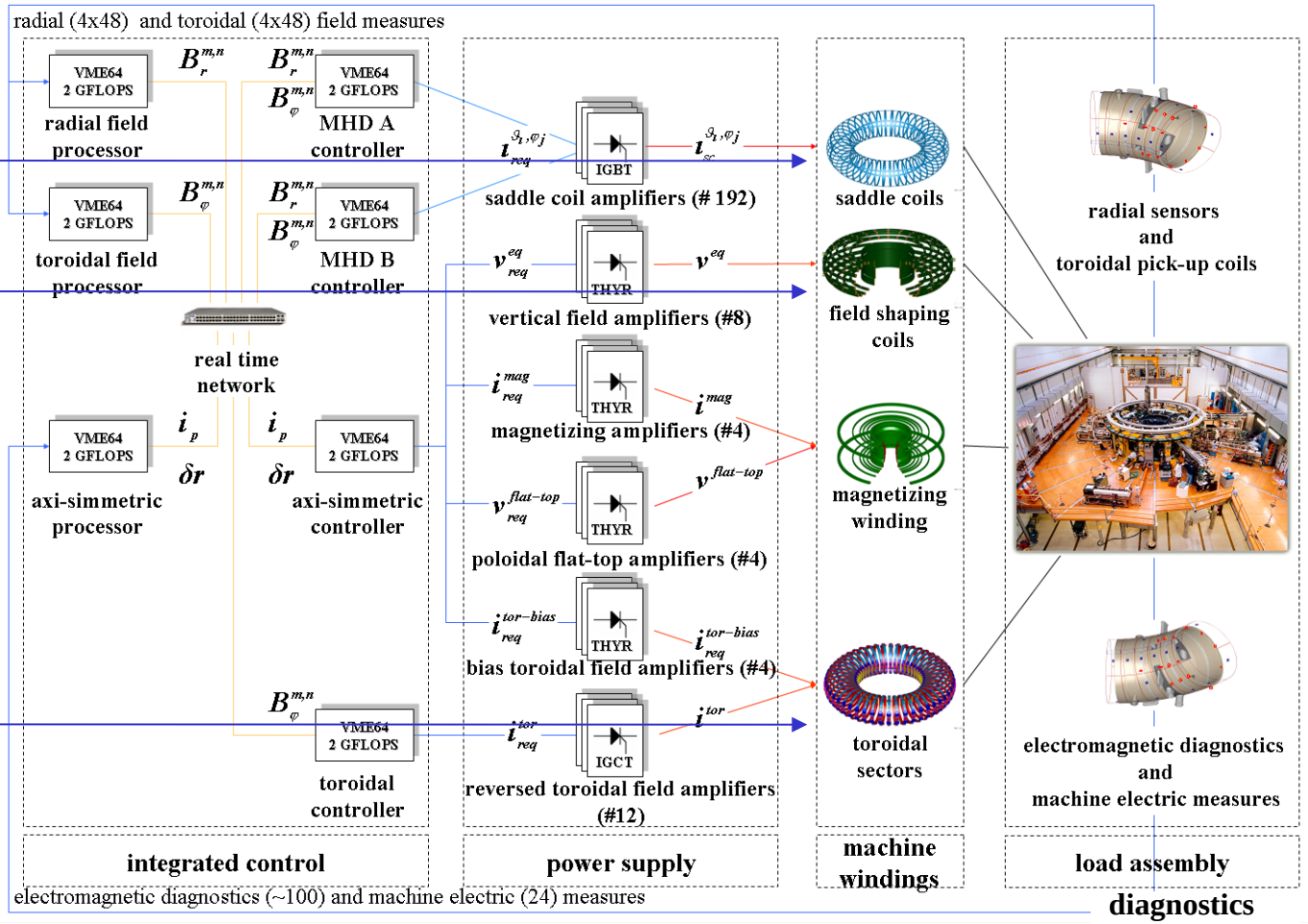
Previous RFX-mod control system

CONSORZIO RFX
Ricerca Formazione Innovazione

MHD mode control

Plasma position control

Toroidal field control



Critical factors in the previous system

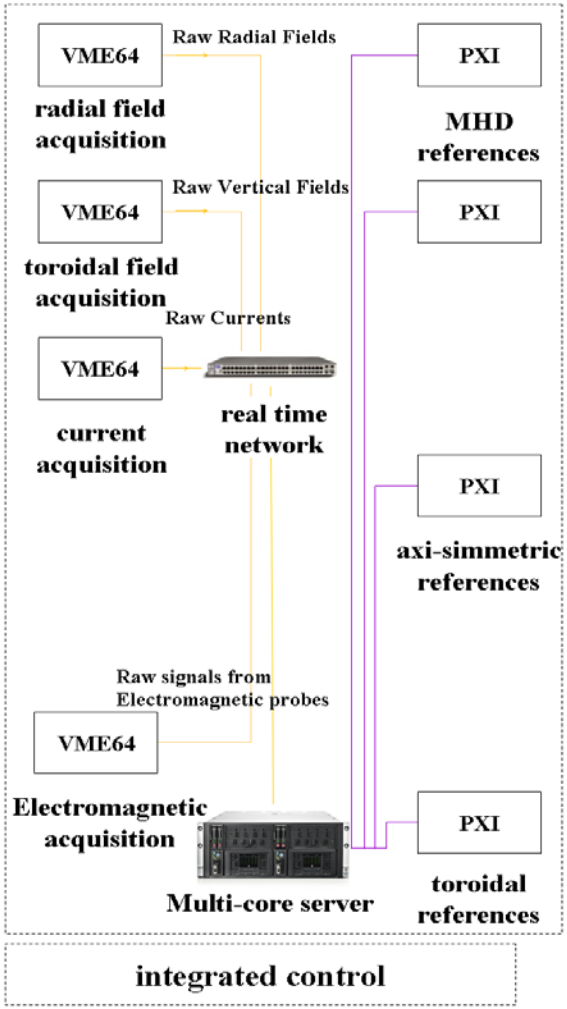
CONSORZIO RFX
Ricerca Formazione Innovazione

- ***Latency***
  - Current latency is around 1.5 ms. This represents a critical factor in quality of control leading sometimes to instabilities.
- ***Sampling frequency***
  - Current sampling frequency is 2.5 kHz. A higher sampling rate improves the quality of integration/derivation.
- ***Computing power***
  - Operations such as sideband correction and sensor radius extrapolation are highly computing-intensive. Currently only most significant modes are considered.
- ***Testability***
  - The possibility of simulated runs of the system would have allowed the detection of bugs in algorithms before running real control.

New Hardware Architecture

CONSORZIO RFX
Ricerca Formazione Innovazione

- Network based data acquisition represents a temporary solution due to budget constraint.
- The use of ATCA ADC boards is foreseen in 2014.
- Tasks carried out by former VME CPUs have been mapped into the server cores.
- The main bottleneck due to communication has been removed



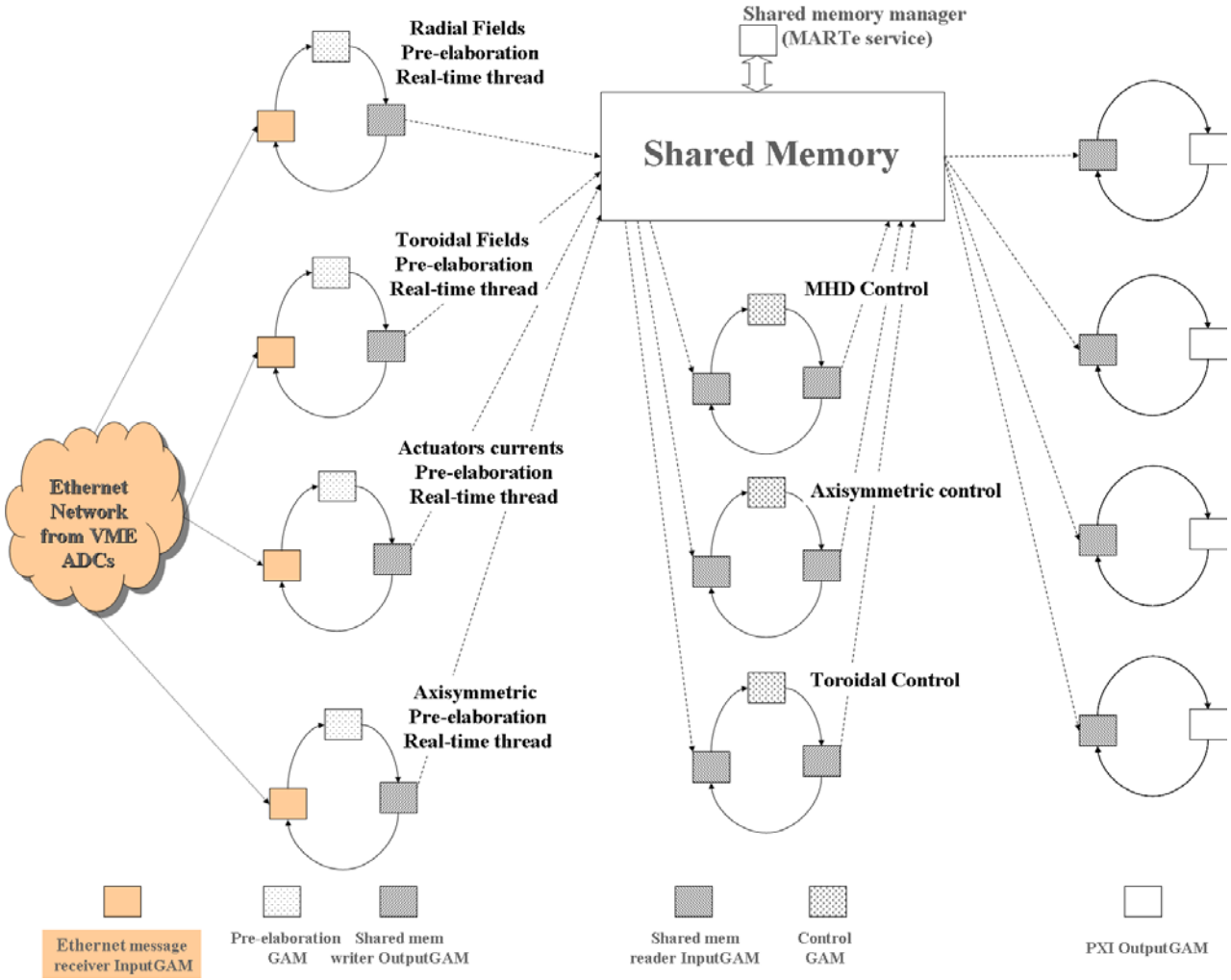
Software framework: MARTe

CONSORZIO RFX
Ricerca Formazione Innovazione

- ***MARTe is a software framework for real-time applications***
  - Originally developed at JET and used for several controls, such as vertical sabilization
- Multiplatform support
  - OS abstraction is carried out by a set of C++ classes
- Single process – multiple threads model
  - Threads are defined in a configuration file
- Agnostic on the kind of computation carried out
  - User provided components extends a generic class GAM (Generic Application Module)
  - Other components implement generic I/O and services
- Configuration specified in a configuration file
  - No changes in code required

MARTe configuration: 11 Threads

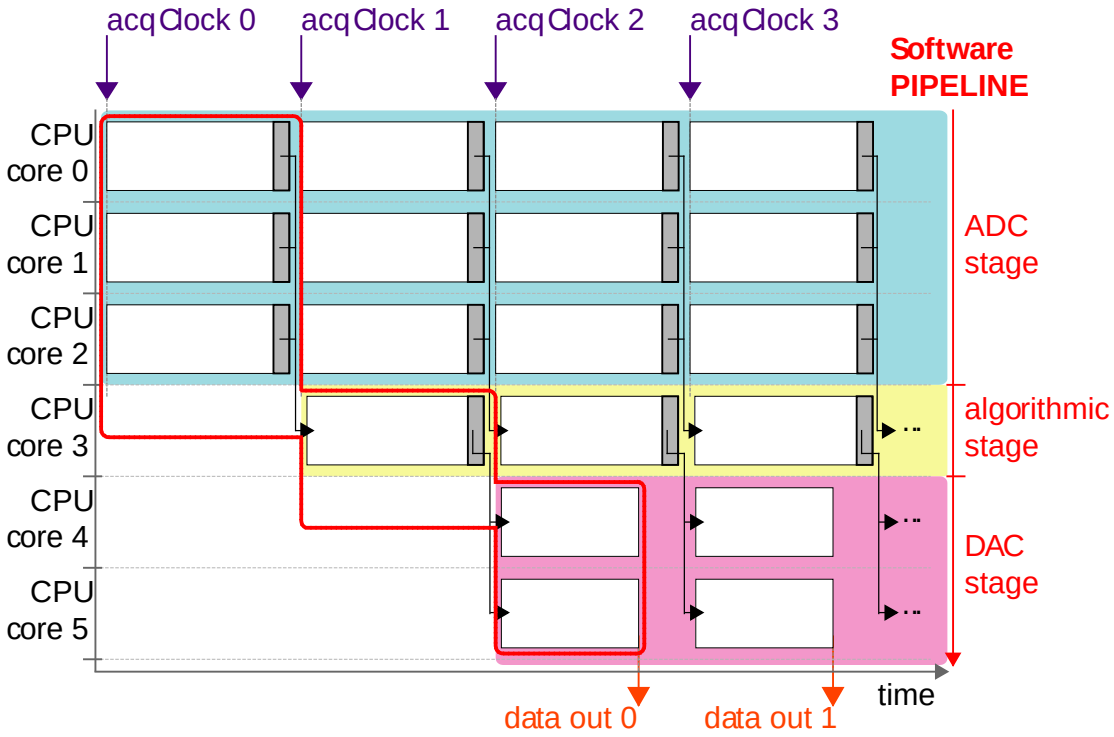
CONSORZIO RFX
Ricerca Formazione Innovazione



Pipelined multicore execution

CONSORZIO RFX
Ricerca Formazione Innovazione

- Pipelined organization with three stages:
  - ***Data Acquisition***
  - ***Control Computation***
  - ***Reference Waveform Generation.***



Linux vs Linux MRG

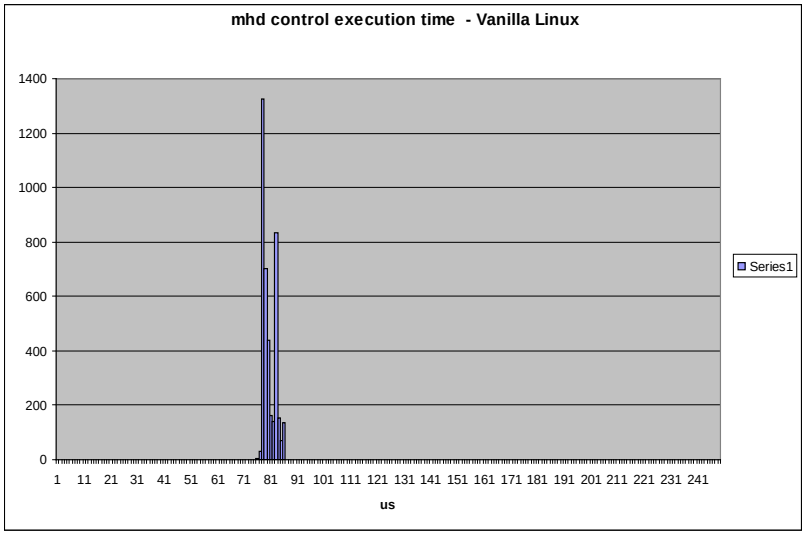
CONSORZIO RFX
Ricerca Formazione Innovazione

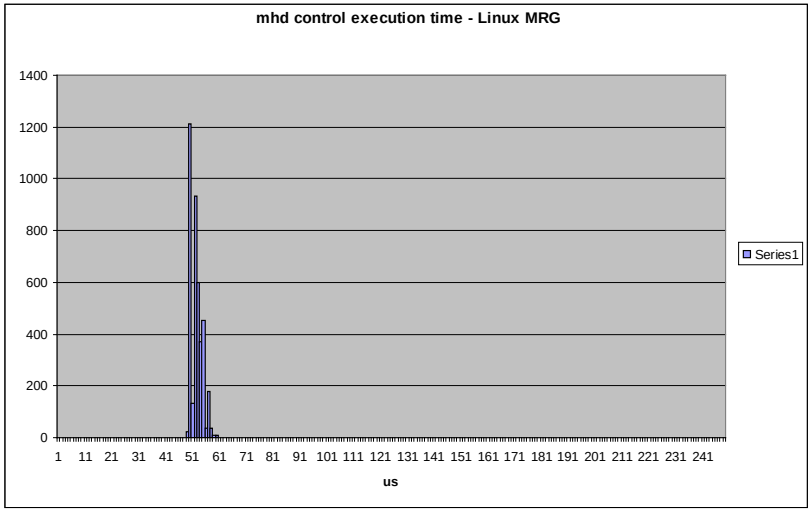
- PREEMP_RT integrated in Linux MRG provides:
  - Preemptible critical sections, protected by rt-semaphores instead of spinlocks
  - Priority inheritance
  - Preemptible interrupt handlers
- All those aspect make the system more deterministic in response
- We expected that advantages could be less evident in multi-core application when contention for resources is reduced

Linux vs Linux MRG

CONSORZIO RFX
Ricerca Formazione Innovazione

- Execution time for control is clearly reduced in Linux MRG even when running on a dedicated core, probably due to a different CPU clock setting
- As expected, jitter is not changed

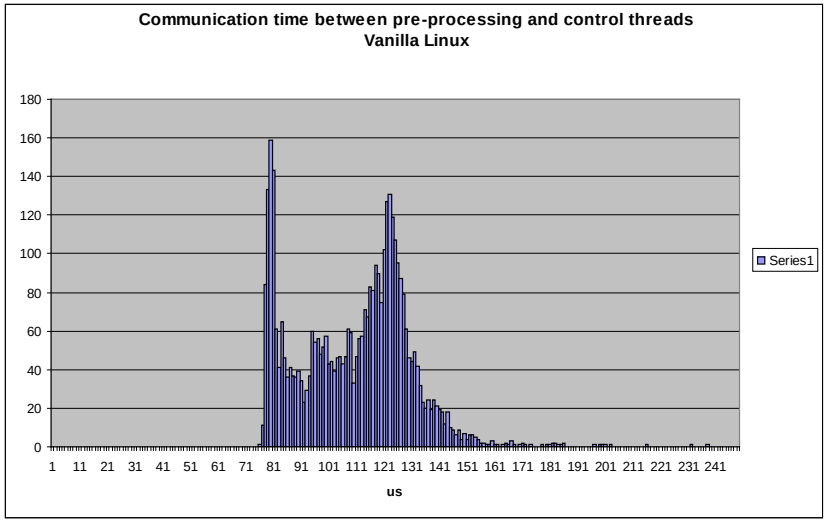


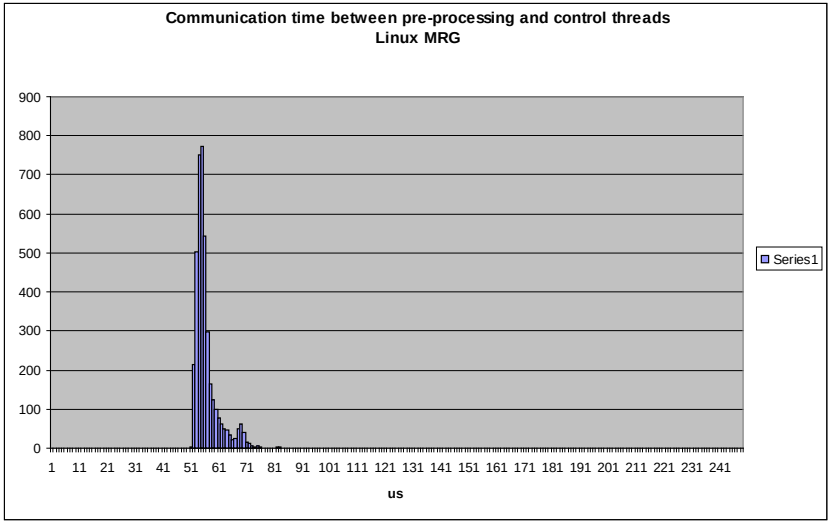


Linux vs Linux MRG

CONSORZIO RFX
Ricerca Formazione Innovazione

- The time required to transfer data from one thread to the other is shown for Linux and Linux MRG
- In this case the scheduler is involved
- Clearly the jitter is largely increased in in respect to Linux MRG

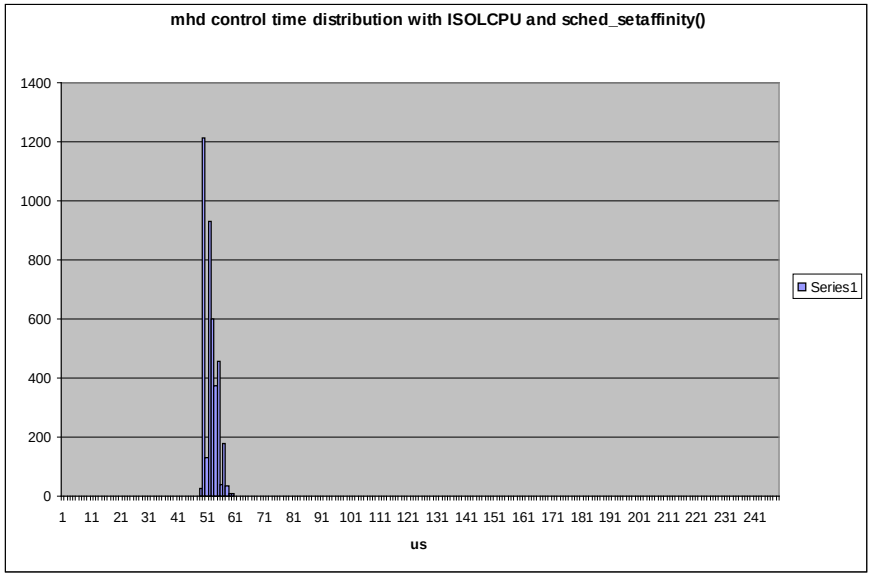


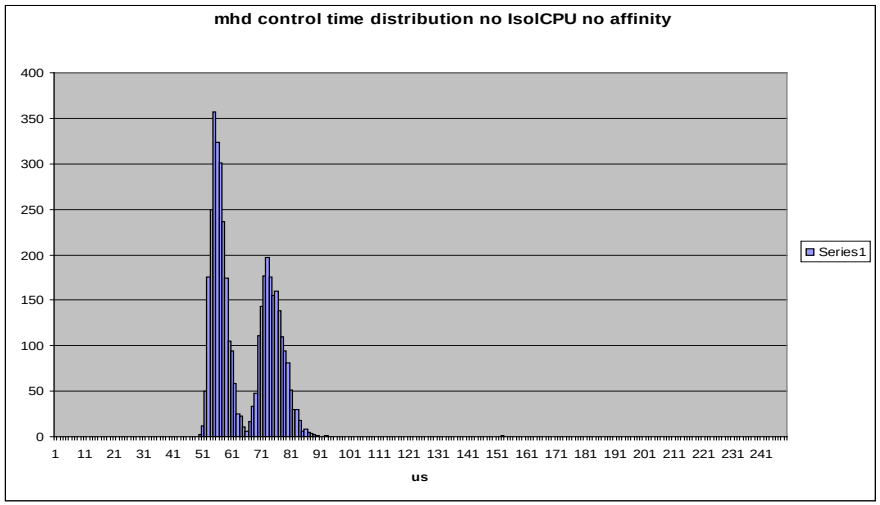


Manual Core Assignment

CONSORZIO RFX
Ricerca Formazione Innovazione

- Assigning threads to cores can be left to the OS Scheduler
- Alternatively manual core assignment can be carried out by the combined usage od ISOLCPU and sched_setaffinity()
- The latter option is mandatory in order to achieve real-time responsiveness

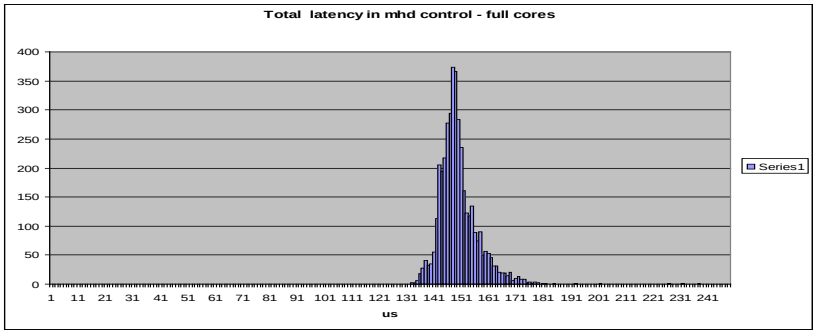


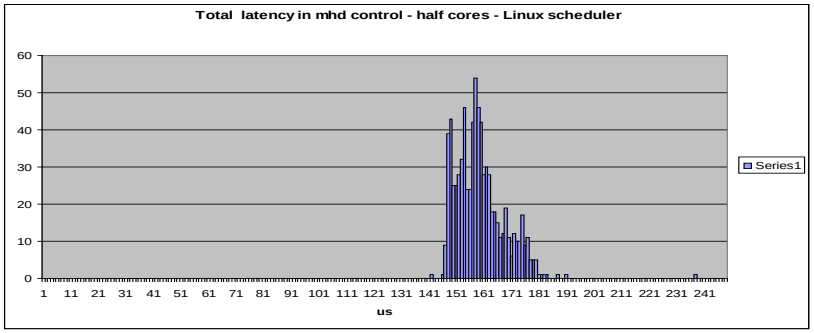


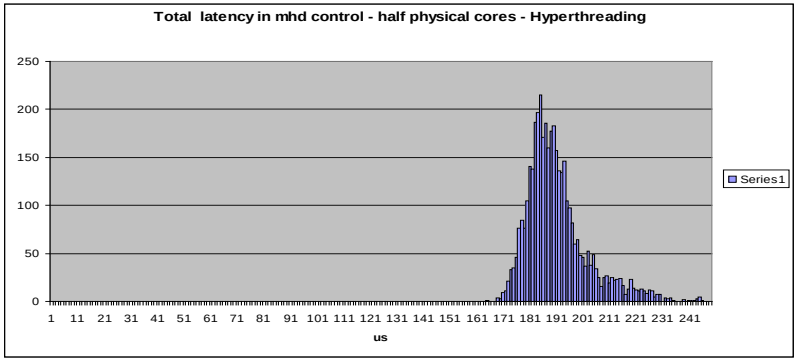
Linux Scheduler vs Hyperthreading

CONSORZIO RFX
Ricerca Formazione Innovazione

- When the number of tasks exceeds the number of available cores, task must be shared
- Two possible approaches:
  - Let the Linux scheduler handle the tasks assigned to each core by a combined usage of sched_setaffinity() and ISOCPU
  - Double the number of “virtual” cores by enabling hyperthreading







Lessons Learnt

CONSORZIO RFX
Ricerca Formazione Innovazione

- The usage of general-purpose hardware allows keeping pace with the mainstream technology evolution;
- The multi-core architecture fits very well with the modular and distributed architecture of the control system;
- The performance of Linux, and especially of its real-time extensions is now comparable with that of proprietary and expensive real-time systems;
- Using a shared software framework avoided re-inventing the wheel and led to a rapid development;
- Among the many positive aspects of MARTe, the possibility of simulating the system proved extremely useful when non IT specialists are involved in the development of the real-time algorithms.