

Industrial informatics

Paolo Burgio
paolo.burgio@unimore.it



UNIMORE
UNIVERSITÀ DEGLI STUDI DI
MODENA E REGGIO EMILIA

High Performance
Real Time **Lab**



**SE NON LO SAI
SPIEGARE
IN MODO SEMPLICE,
NON L'HAI CAPITO
ABBASTANZA BENE**

- Einstein



THE EXAM

Mandatory written test

- › A mix of multiple (closed) responses, and open questions
- › Up to 30/30
- › 3 dates, 1 of which in December (tentative). See Q&A on website

Optionally, to improve your mark you can choose between

- › Oral (3-4 questions)
- › Project (recommended for internship and theses)
- › (typically $+4/-\infty$)

Course material

Course website

https://hipert.unimore.it/people/paolob/pub/Industrial_Informatics/index.html

Course slides

- › Available on Moodle, early preview on github
 - <https://github.com/HiPeRT/IndInf.git>
- › Typically, enough to pass the exam, also if not attending class
- › Hands-on exercises



Textbooks

- › See course website
- › Add reference at the end of each slides block





Required skills

Unix Programming

- › Also win is fine..
- › C/C++ preferred

A bit of electronics might help...but it's not mandatory

Passion, passion, passion!!



Required material on your side

A laptop with GNU/Linux

- › For fast prototyping
- › Also win is fine...you can use Cygwin or Windows Subsystem for Linux (WSL)

Embedded boards for hardcore hands-on

- › Raspberry Pi
 - You should already have it for the IoT course
- › Arduino (do you want this?)
- › NodeMCU
 - ESP8266/ESP32 processor
- › ...plus some basic stuff (breadboard, electric cables, leds...)

CAD tools

- › CODESYS by 3S-Smart Software Solutions GmbH
 - I will use V3.5 SP10 Patch 3+, better having the same
 - I can provide it, as well as textbook



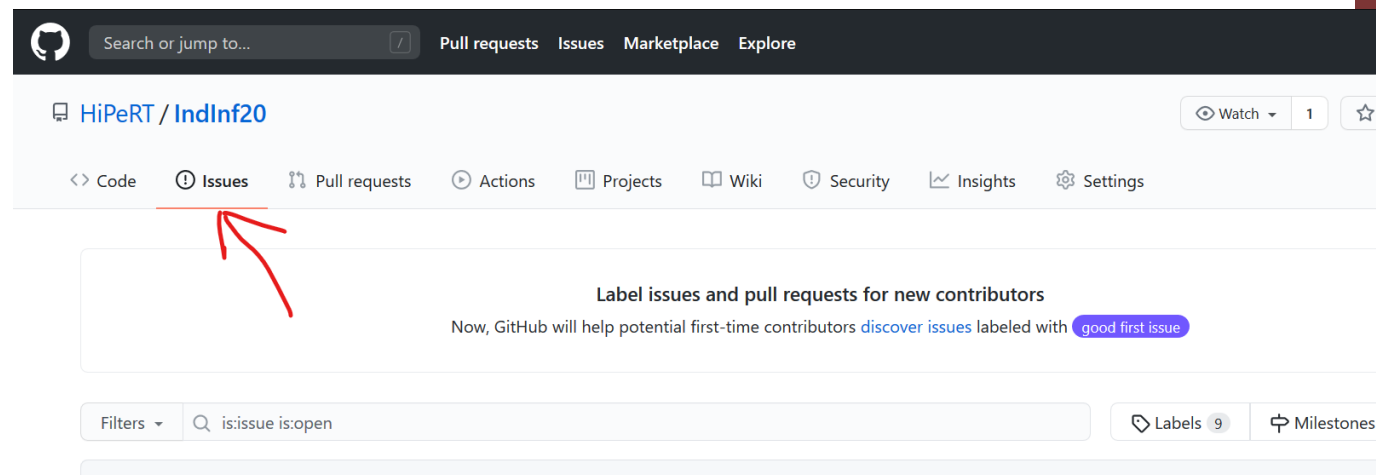
How to contact me?

AKA: ricevimento

› paolo.burgio@unimore.it

But let's try something different..

- › For every question, open a ticket ("issue") on GitHub
 - <https://github.com/HiPeRT/IndInf.git>
- › So, all of your colleagues will enjoy the answers
- › Netiquette: before asking, search in "issues"





What is this about?



Why *"Industrial"*informatics?

What makes them different by "standard" informatics?

- › Applications
- › Computers
- › Requirements
- › Frameworks
- › Processes (industrial, not "OS processes" 😊)
- › Technologies



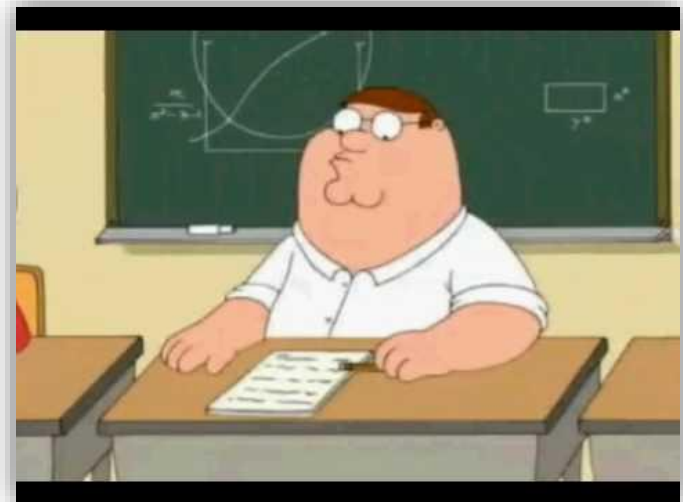
"domain driven"...?

Es: industrial plants are different than biotech, or nuclear plants

- › Tens of application domains..
- › Is automotive also industrial...today? (rhetorical)
 - And aerospace? Planes, ships, surveillance drones?
 - Safety critical systems
- › Rise of many-core computers
 - Size, Weight and Power constraints - SWaP
- › Internet 4.0 – Internet of Things
 - How can we use Machine Learning?

Past, present and future

- › Not easy to teach, not easy to structure





What's common?

› Tight interaction with the environment

- A plant, a machine
- Trough sensors and actuators
- A lot of computers
- Huge cost
- Centralized controller & data analytics w/human



- *Eco-System architecture*
- *Cyber-physical systems*
- *Embedded systems*

› We want guarantees

- Worst-case timing behavior
- Often, safety
- Reliability, fail-tolerancy, also in monitoring
- Security (we won't see this)



- *Real-Time systems*
 - *Dependable*
 - *Observable*
- (WON'T SEE THIS)*

› Compliancy with previous technology

- Lombardia, ER are highly-industrial areas (consider also farms)
- Can't always use "startup-like" models
- Not easy to replace legacy HW/SW



- *Old technologies*
- *Legacy software*
- *Old processes*

› The ~~future~~ present



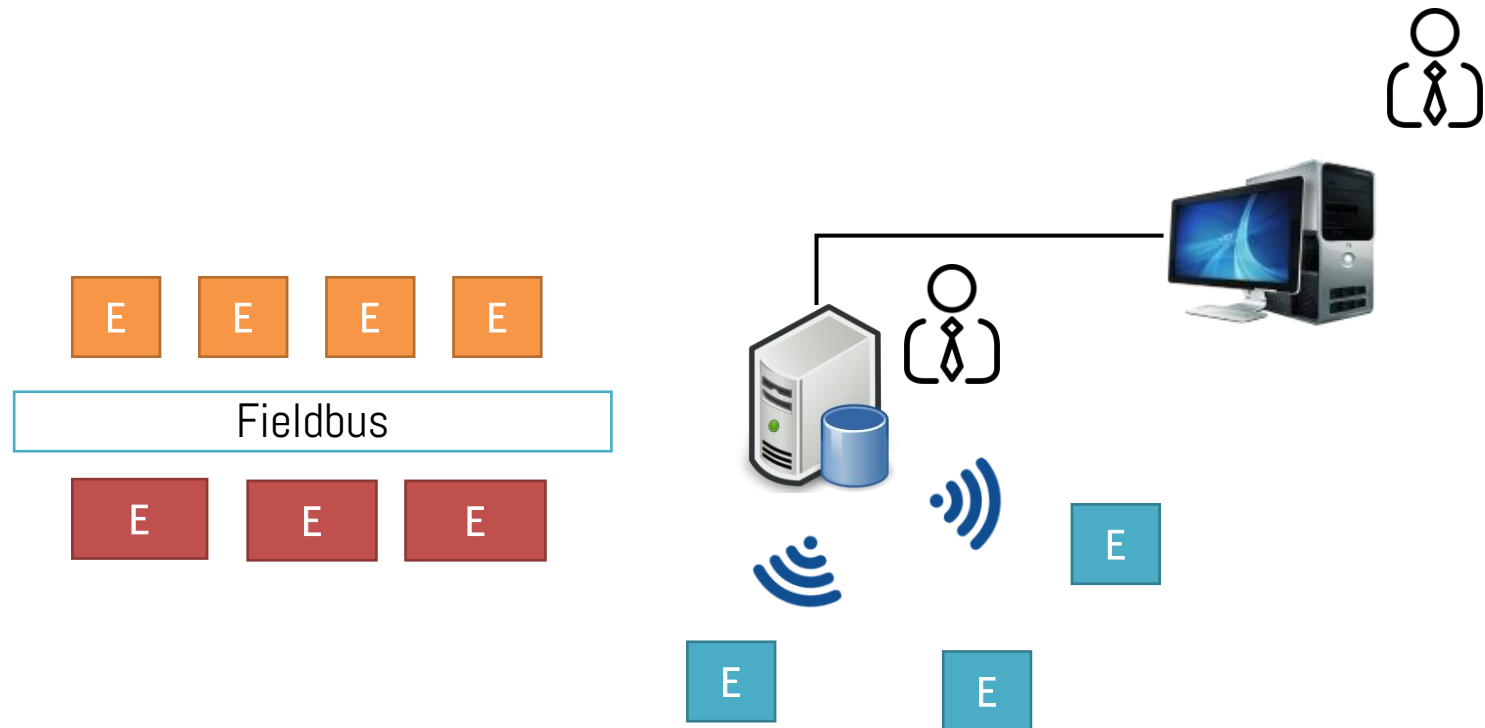
- *Machine Learning*
- *Big data*
- *IoT*
- *...?*



CPS - Cyber Physical (Eco)Systems

Tight interaction with the environment

- › Typically, multiple embedded computing units close to the plant
- › And one (or more) centralized controller
- › Today, these computational edges have more and more computing power





Why software engineering?

Programmers' work: creating software

- › Coding (hopefully well)
- › Creating Uxs
- › Hacks to make things work....

Engineers' work: designing software

- › Maintainable
- › Well documented
- › Scalable – **what does this mean?**



Why software engineers?

Expectations



Reality

- › Bridging the gap between devs, QAs, management...



Our journey

Will cover these main topics

- › Collaborative tool (quickly)
- › Languages & grammars, Automations & machines
- › Computing architectures: PLCs, GPGPUs, reconfigurable architectures, ...
- › Software engineering
- › Programming industrial systems: PLC, embedded GPGPUs & CUDA

..and...

- › A glance on embedded machine learning
- › Automotive systems
- › 2-3 seminars, chosen together

...always with an hands-on approach!



References



Course website

- › http://hipert.unimore.it/people/paolob/pub/Industrial_Informatics/index.html

My contacts

- › paolo.burgio@unimore.it
- › <http://hipert.mat.unimore.it/people/paolob/>
- › <https://github.com/pburgio>