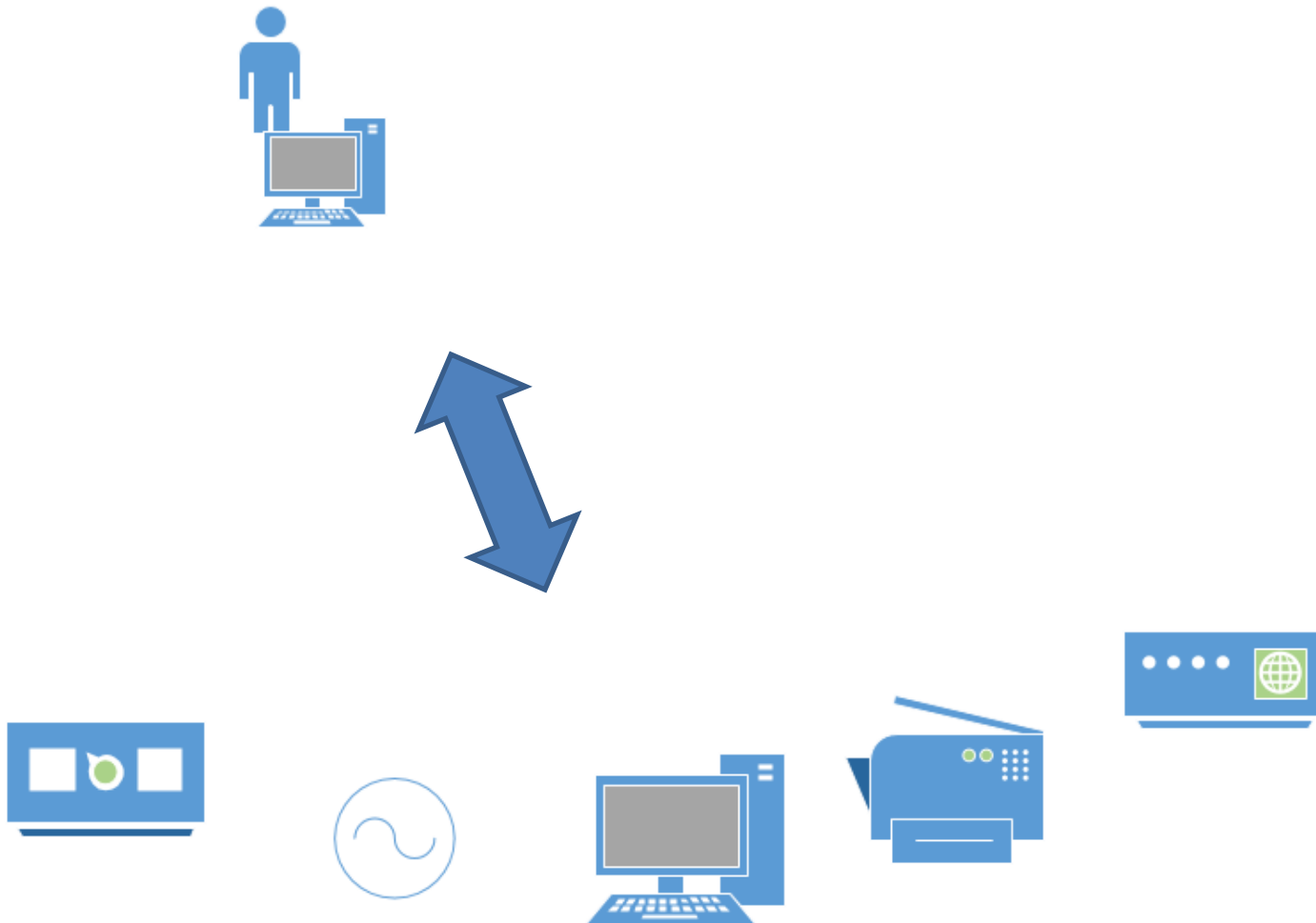


What is "Control System" or "Framework"?

General purpose of control system

- need to control **a lot of** equipment from **remote**



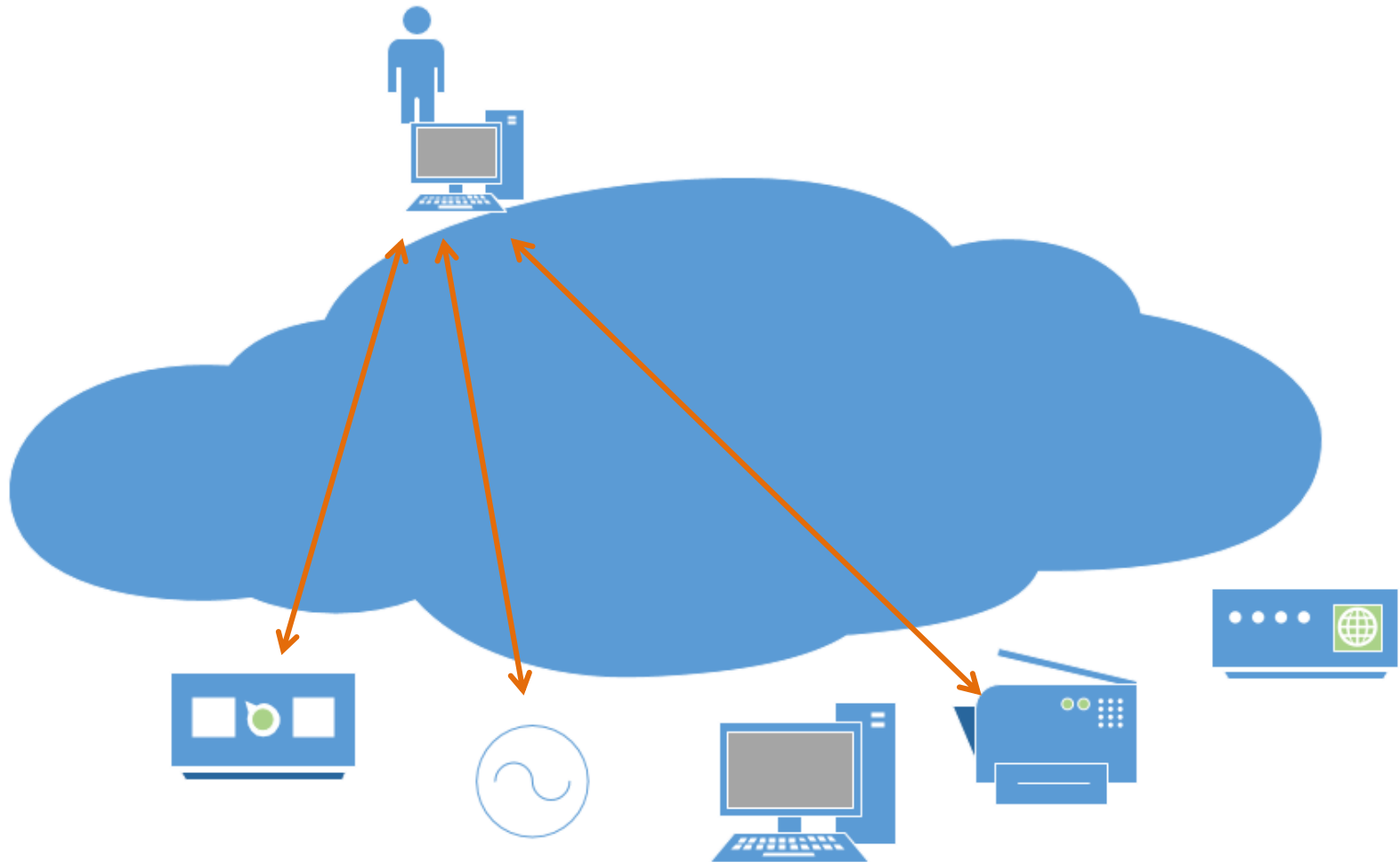
What is "Control System" or "Framework"?

We need a "Network" to connect instruments and controller



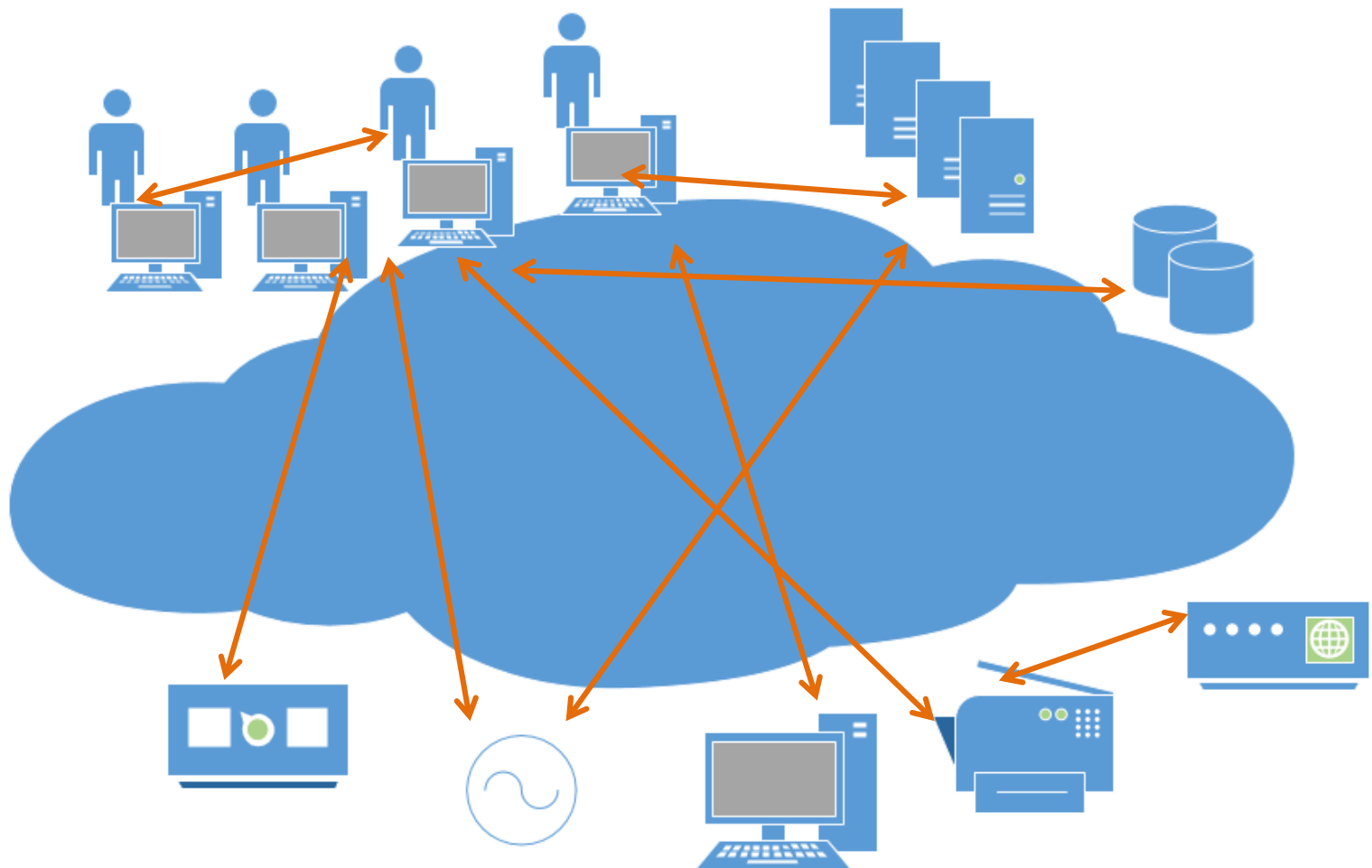
Small system

In case the number of target instruments are limited, simple program can do everything. This is true for small laboratory.



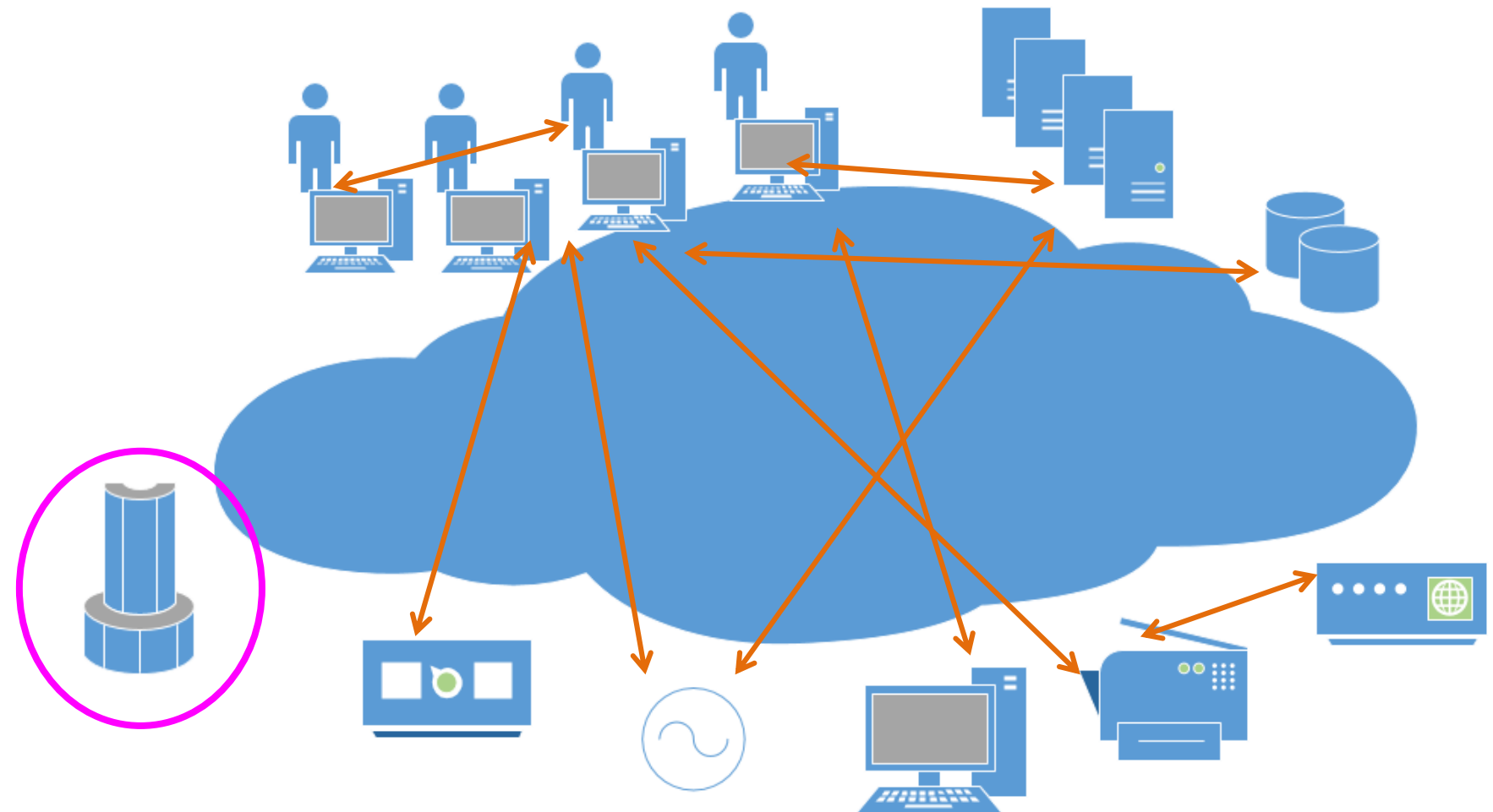
Control Framework

We need a control framework in case many instruments, programs, developers, operators, Tight **communication** and **organic linkage** between components are required.



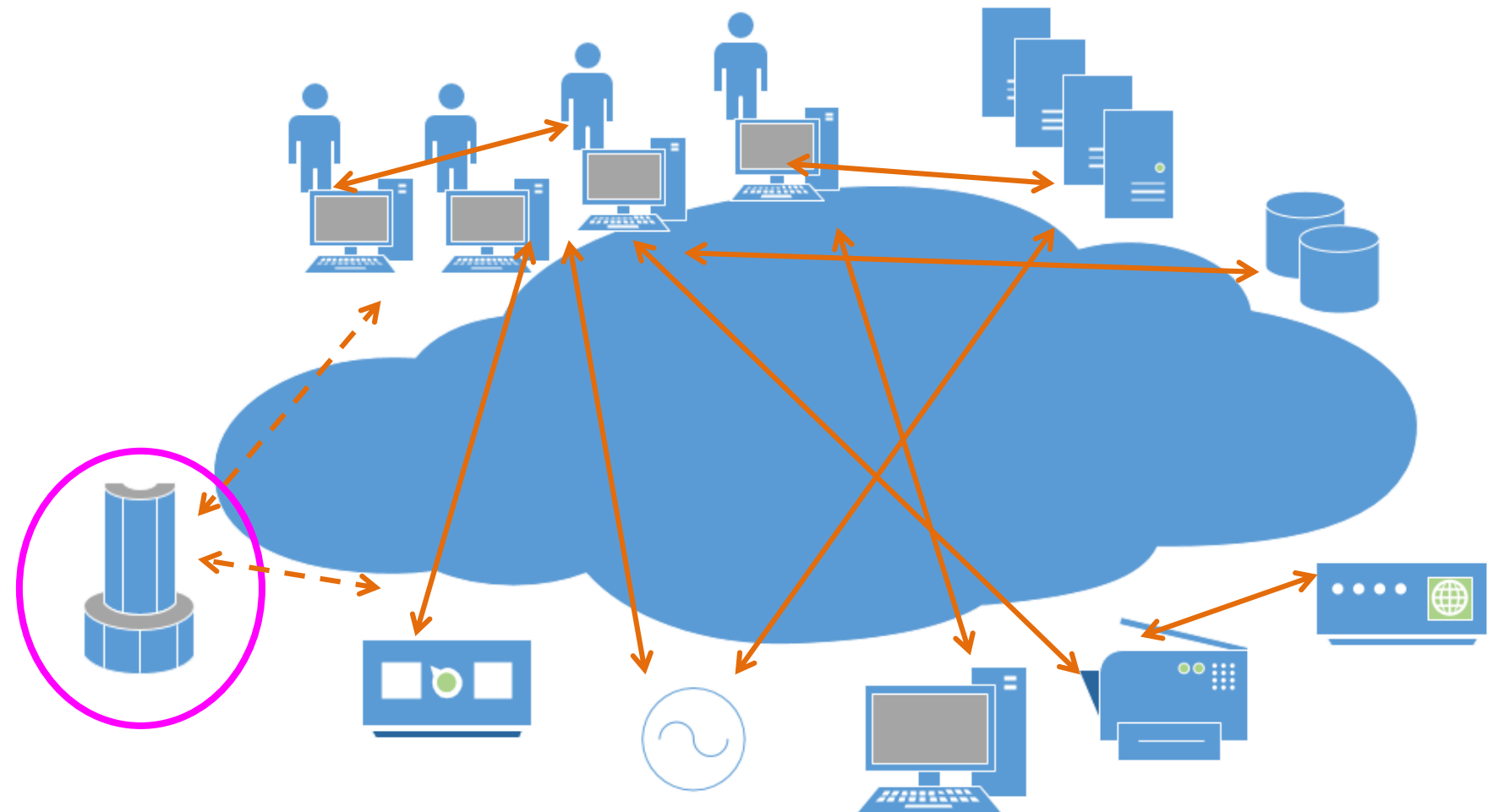
If we want to add a "New" instrument

Need same "Language" and "Communication Protocol"
with existing environment.



In reality

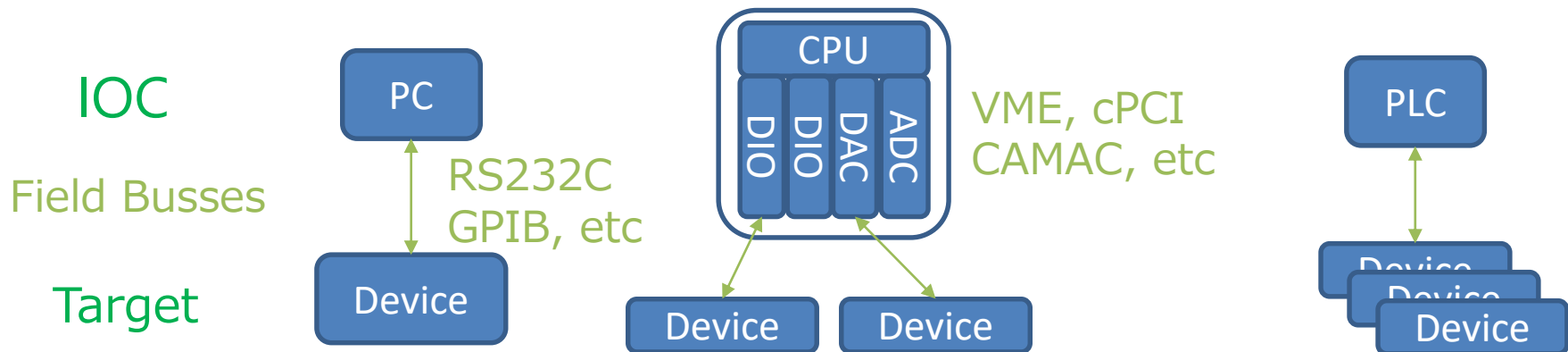
Commercial product does not support EPICS or Tango framework in many cases. We need to develop support them.



In case of EPICS

IOC (Input/Output Controller) manage hardware

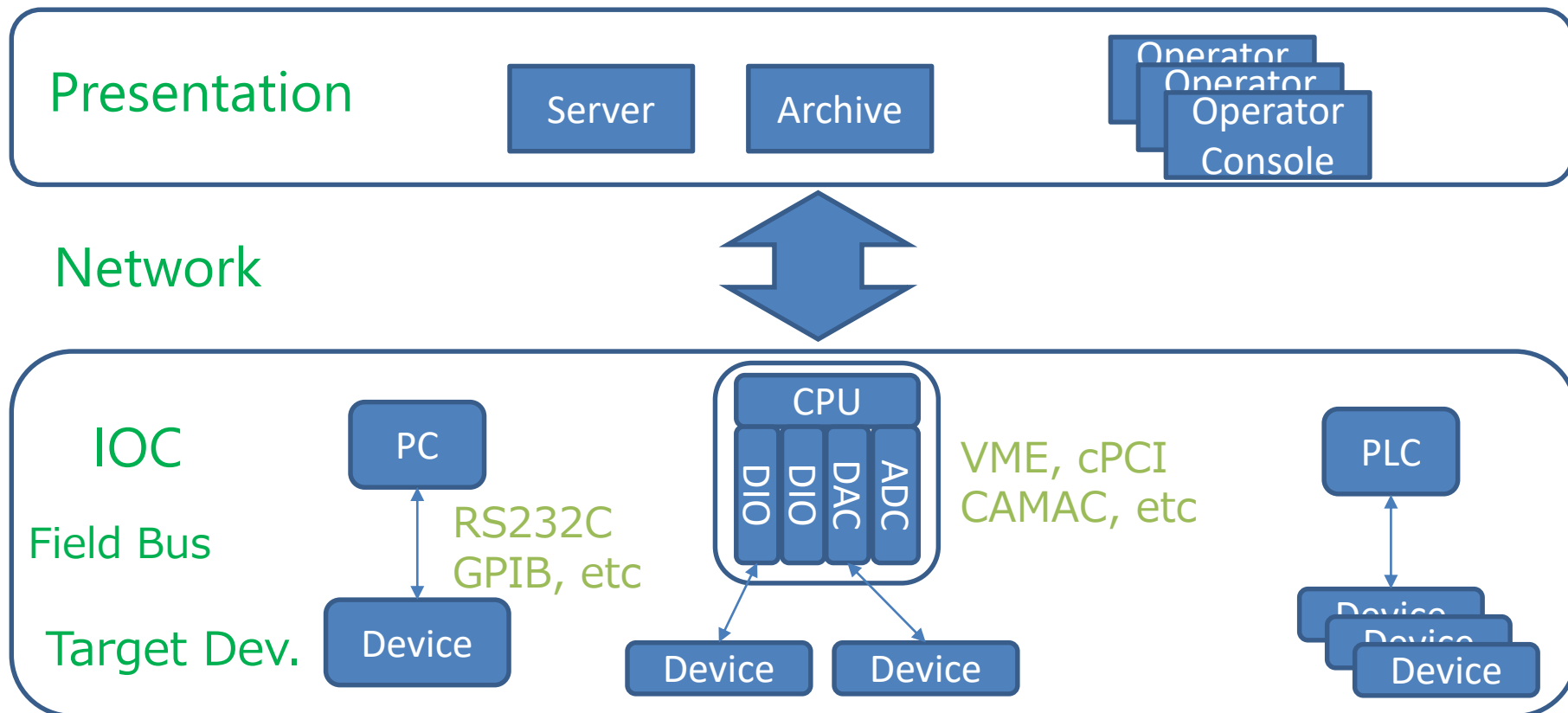
- IOC: VME, PC, PLC, software, etc...
- Need special code **depends on** each device



EPICS : Communication

Need to communicate with standard protocol between presentation layer and IOC layer.

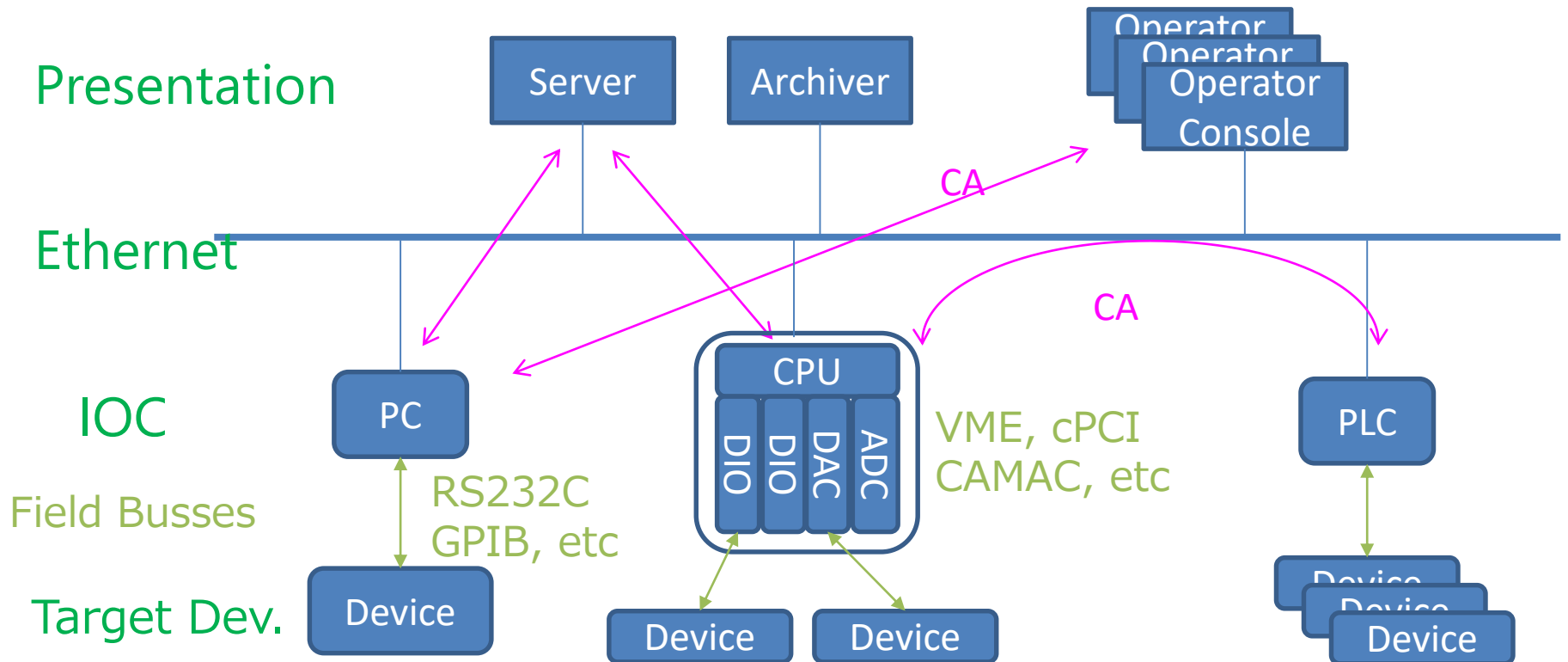
- Presentation Layer : Server process, Archiver, GUI, etc
- **Independent** from each device



EPICS : Communication Protocol

Channel Access protocol (PV, Process Variable)のやりとりをする

- Network transparent. Distributed system.
- CA protocol is used between IOCs



When you buy an instrument on the market

If the device has an embedded ioc, we have nothing to do!



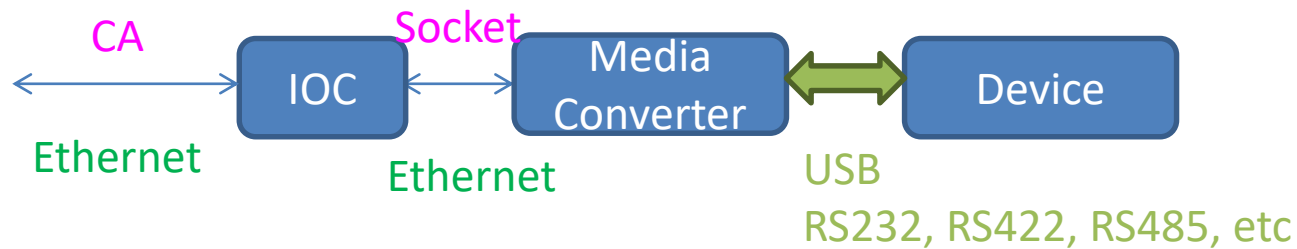
Just power on, plug to the network, and the instruments speak/listen CA protocol

In reality, very few instruments understand EPICS or Tango. Accelerator research is very small field on the market.

for example : RS232C device

Media converter (RS232C \leftrightarrow Ethernet) are often used

- Need to put media converter near to the device
- Distance between IOC and Device can be very long

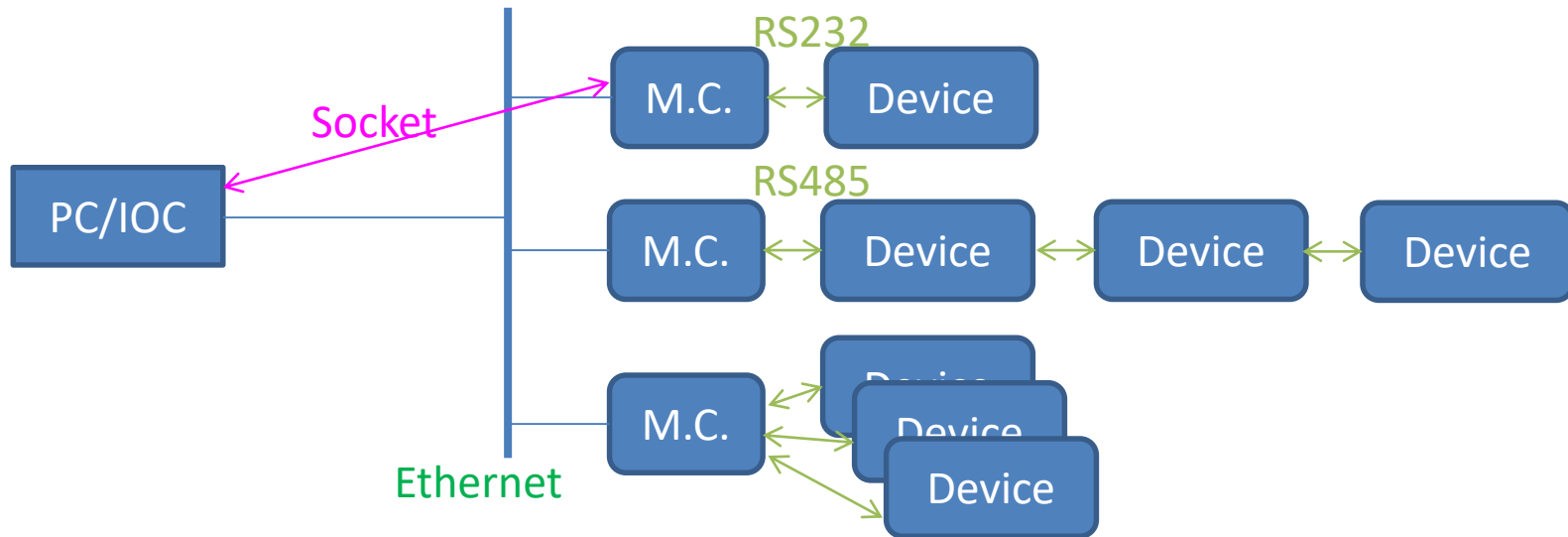


Socket protocol are used between IOC and M.C.

“Steam Device”

Example commercial product

- Serial-Ethernet Converter (Media Converter)



Many company produces similar product.
So many variety of the price and functionality.
2, 4, 8, 16 port, etc.



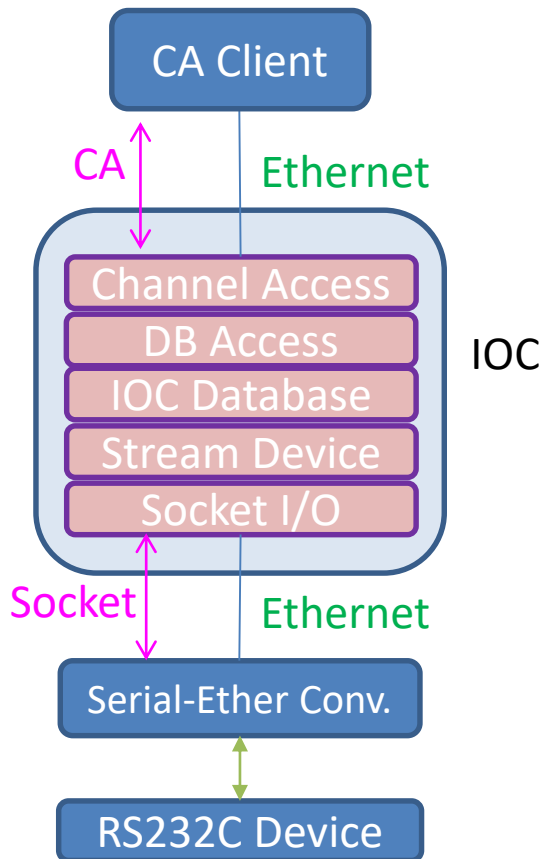
Photo from <http://japan.moxa.com/>

EPICS + Media Converter

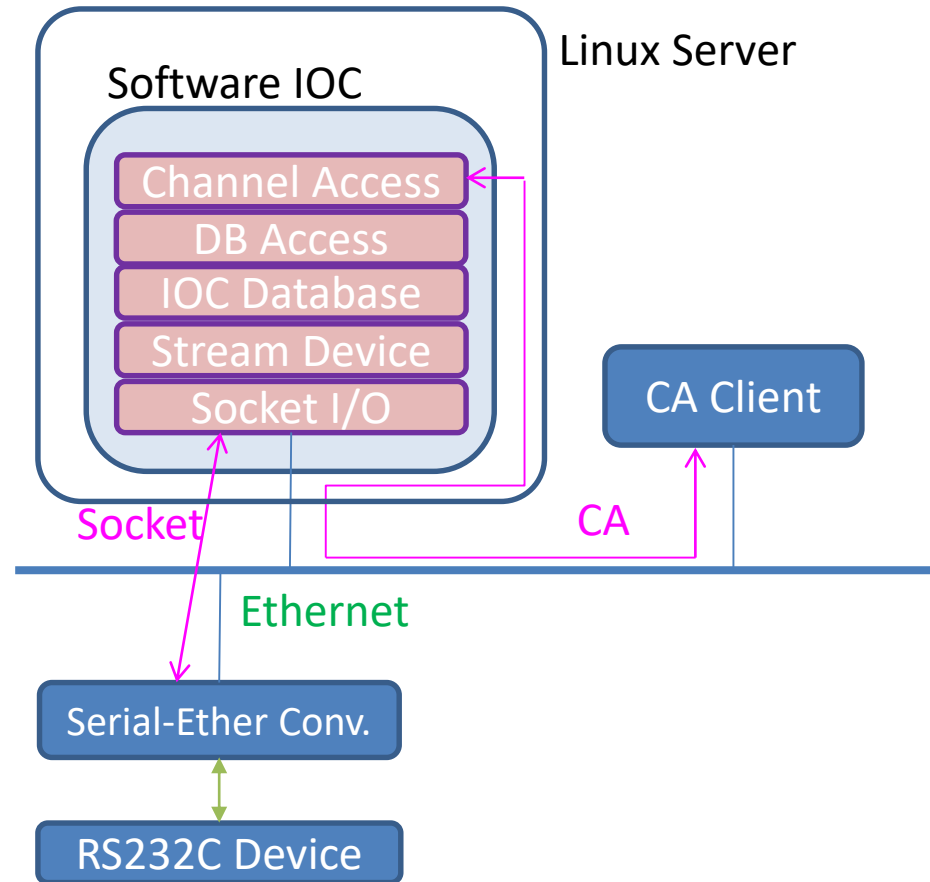
soft IOC on the server computer.

- IOC is a kind of "Protocol Converter"

Logical Description

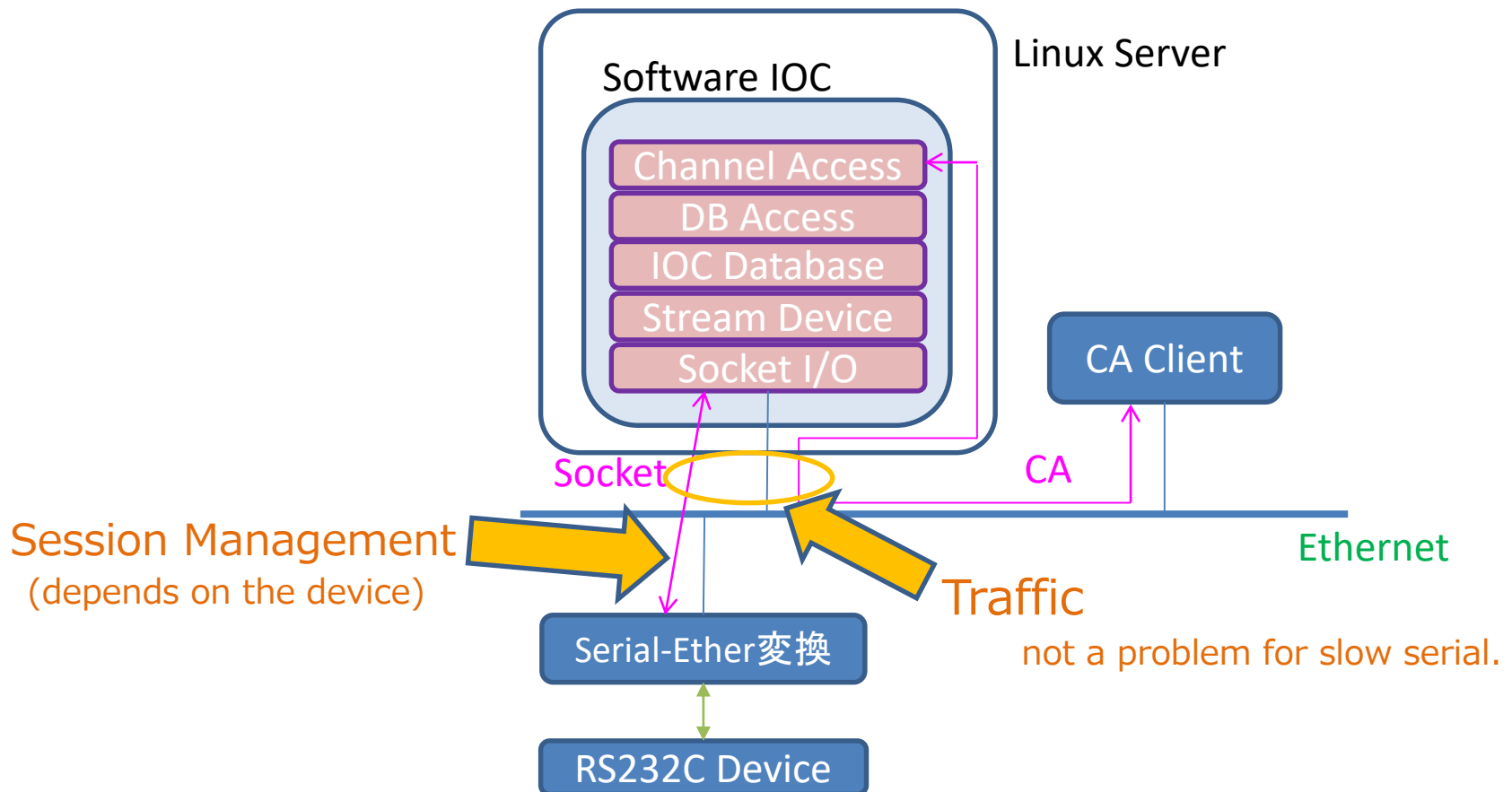


Physical Description



Disadvantage of the media converter

- Tend to increase network traffic
- You need to manage a session between IOC and media converter.
 - For example: Initialize Socket protocol, reconnect, etc
 - You don't need to care for CA protocol. (already defined)

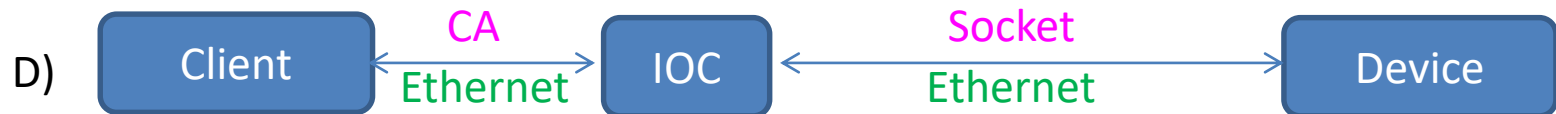
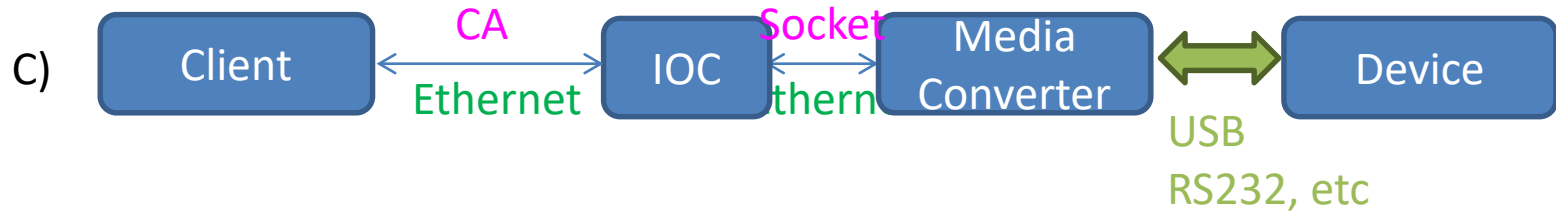
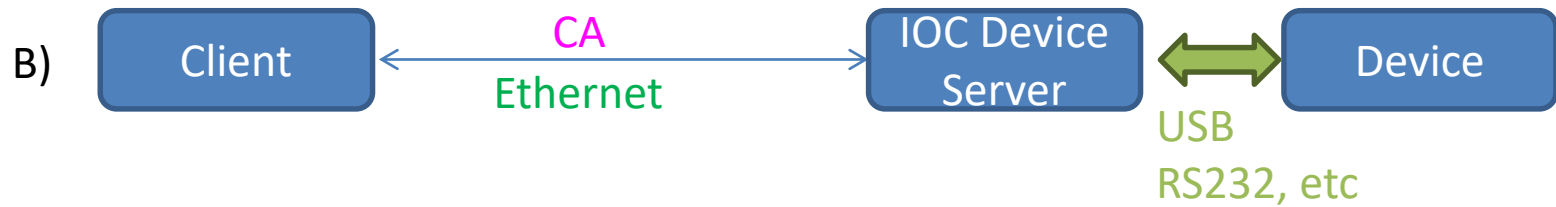


not an ideal solution, but better than media converter

We want a “small” device server attached to the device



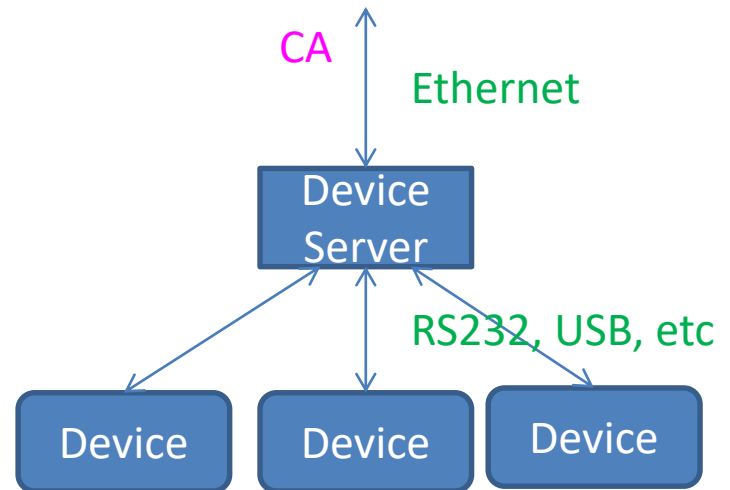
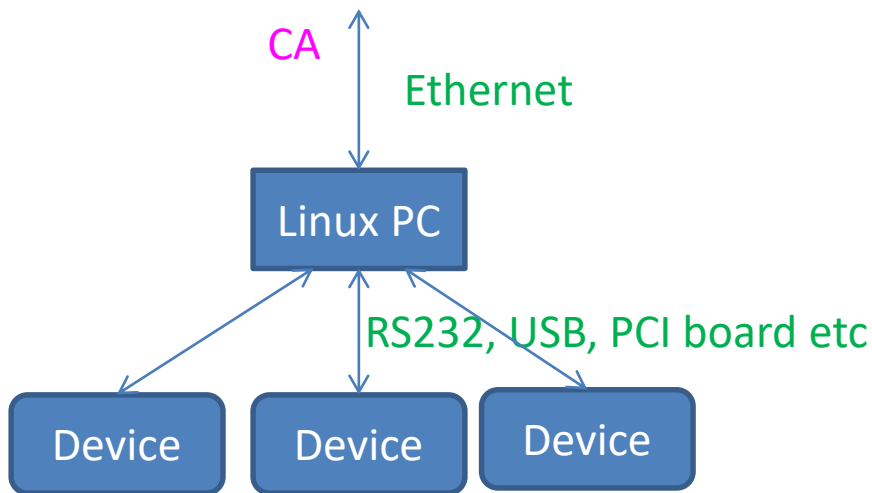
Comparison of configuration



wait...

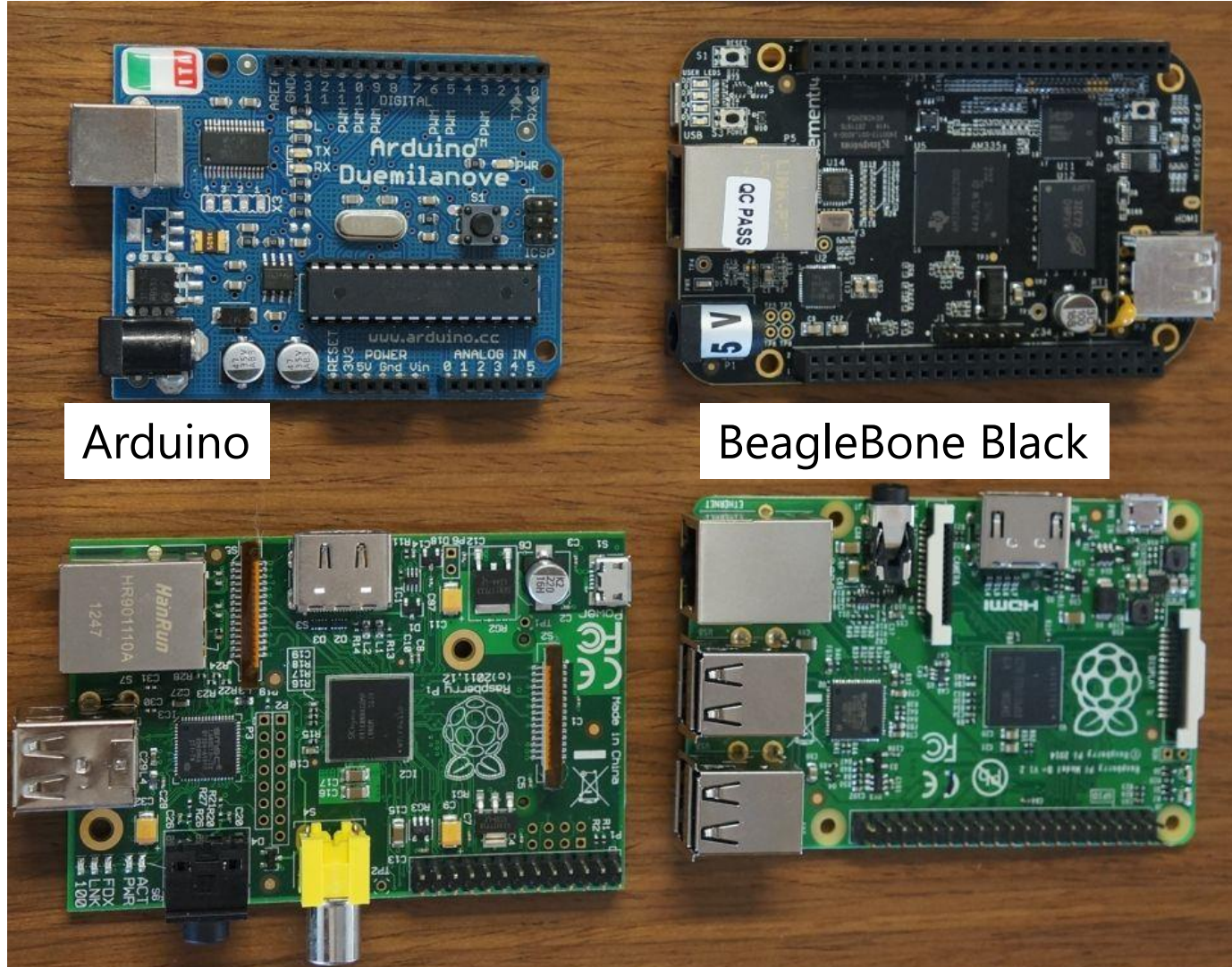
Is there any differences between Linux PC and Device Server?

- Basically no difference
- Price, Size, Power consumption, TCO (Total cost of ownership), etc
- Hardware support for long time
- Consider these requirement to **select the base hardware**



Candidates

Credit-Card size board



Arduino

BeagleBone Black

Raspberry Pi (model B)

Raspberry Pi (model B+)

Candidates (1) [RasPi 3 is not listed in this table]

	BeagleBone Black(Rev.C)	Raspberry Pi (ModelB)	Arduino Uno
CPU	TI AM3359 (ARM Cortex-A8)	Broadcom BCM2835 (ARM11)	Atmel ATmega 328
Speed	1 GHz	700 MHz	16 MHz
RAM	512MB DDR3L@400MHz	512MB SDRAM@400MHz	2 KB
Storage	Onboard eMMC 4GB microSD slot	microSD slot	32 KB Flash
I/O(GPIO)	65	8	20
ADC	7	n/a	8
Ethernet	10/100 x1	10/100 x1	n/a
OS	Debian(default) Angstrom Linux Fedora etc	Raspbian(Debian) Pidora(Fedora) ARCH linux etc	n/a
Size [mm]	86.4 x 53.3	85.6 x 54	75 x 53.3
Price	¥ 6,500	¥ 3,940	¥ 3,024