

Fieldbusses



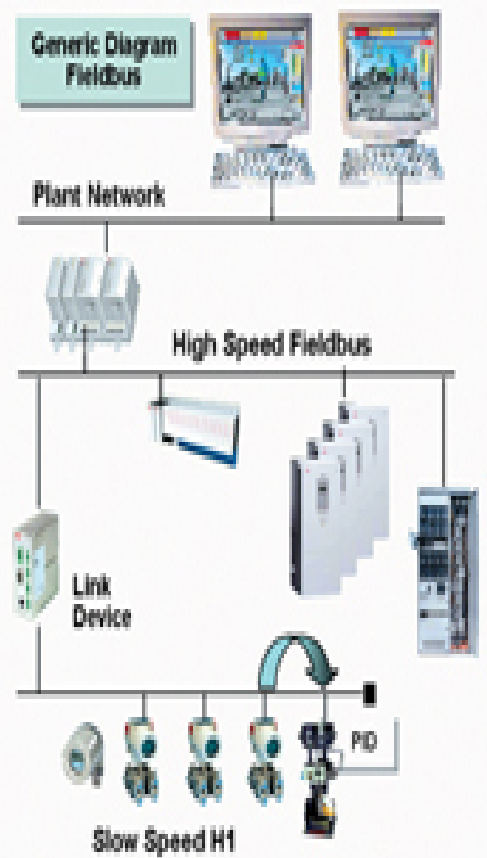
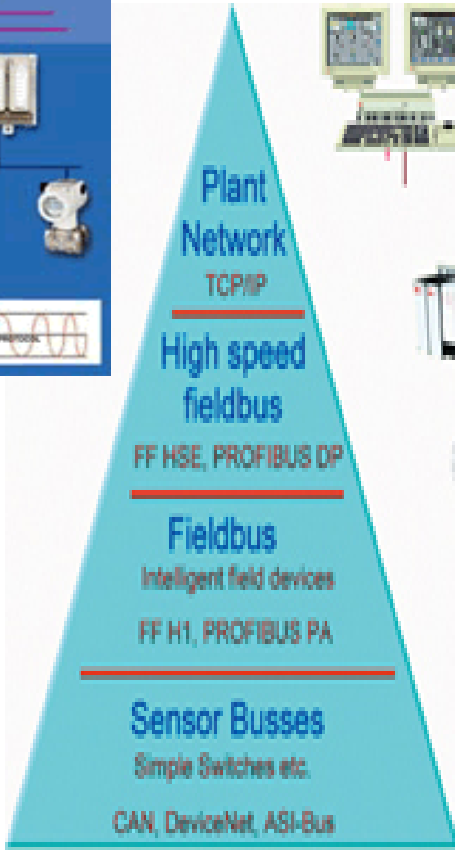
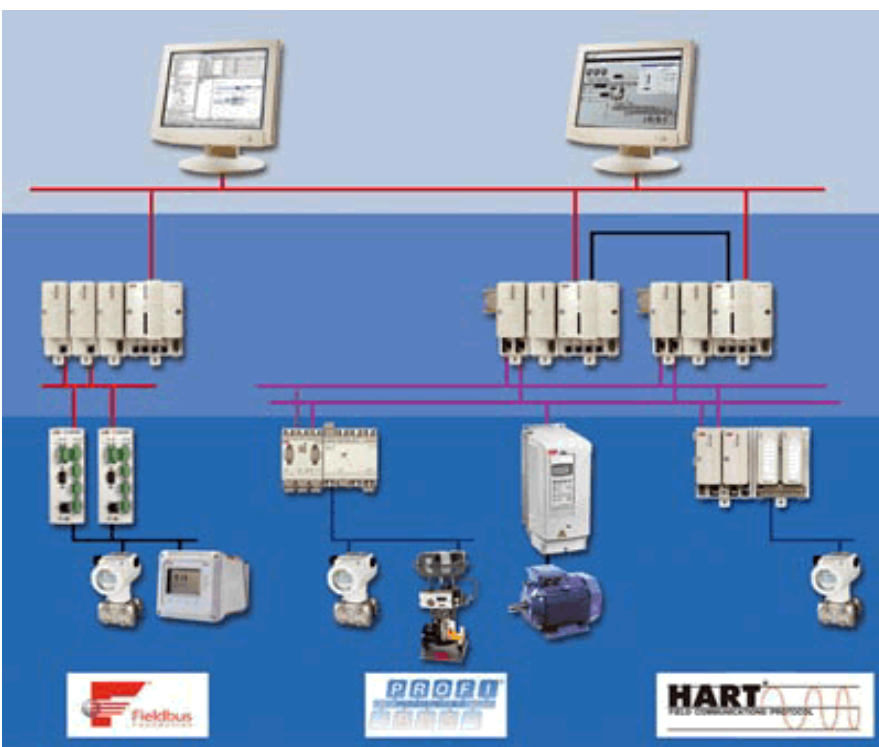
awaiting ...

- Fieldbus is the name of a family of industrial computer network protocols used for real-time distributed control,
...
- standardized as IEC 61158.
 - (might be “conflicting” with IOT)
- private company versions exists
- but interoperability is a “must”
- as well as safety, security, fault tolerance,...
- The five nines: 99.999%

- Fieldbus is the name of a family of industrial computer network protocols used for real-time distributed control,
...
- standardized as IEC 61158.
 - (might be “conflicting” with IOT)
- private company versions exists
- but interoperability is a “must”
- as well as safety, security, fault tolerance,...
- The five nines: 99.999% uptime \approx 5 minutes downtime a year !

- IEC 61158 consists of the following parts, under the general title Digital data communications for measurement and control – Fieldbus for use in industrial control systems:
 -
 - Part 1: Overview and guidance for the IEC 61158 series
 - Part 2: Physical Layer specification and service definition
 - Part 3: Data Link Service definition
 - Part 4: Data Link Protocol specification
 - Part 5: Application Layer Service definition
 - Part 6: Application Layer Protocol specification
- *and the standard is expensive ...*

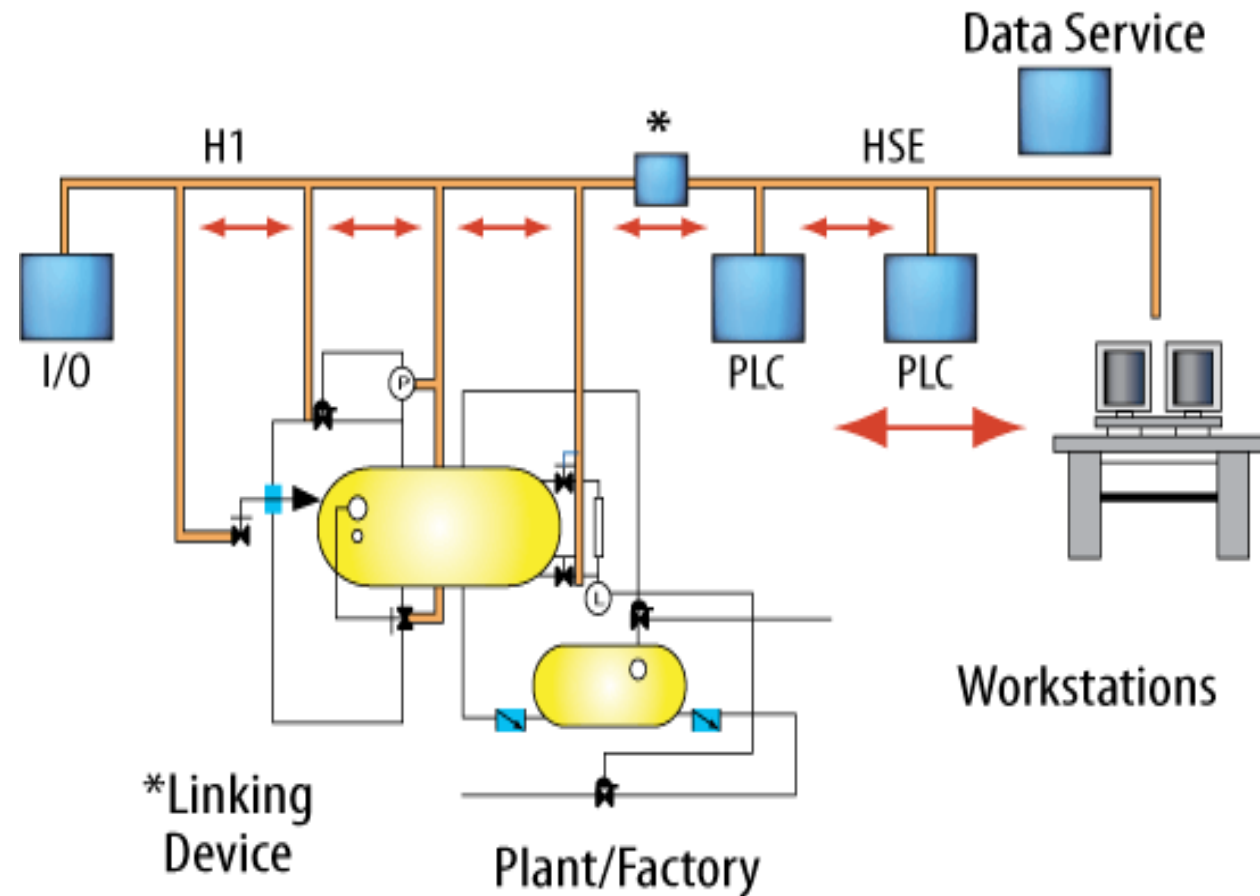
Hierarchy / concepts



Fieldbus ⇅	Bus power ⇅	Cabling redundancy ⇅	Max devices ⇅	Synchronisation ⇅	Sub millisecond cycle ⇅
AFDX	No	Yes	Almost unlimited	No	Yes
AS-Interface	Yes	No	62	No	No
CANopen	No	No	127	Yes	No
CompoNet	Yes	No	384	No	Yes
ControlNet	No	Yes	99	No	No
CC-Link	No	No	64	No	No
DeviceNet	Yes	No	64	No	No
EtherCAT	No	Yes	65,536	Yes	Yes
Ethernet Powerlink	No	Optional	240	Yes	Yes
EtherNet/IP	No	Optional	Almost unlimited	Yes	Yes
Interbus	No	No	511	No	No
LonWorks	No	No	32,000	No	No
Modbus	No	No	246	No	No
PROFIBUS DP	No	Optional	126	Yes	No
PROFIBUS PA	Yes	No	126	No	No
PROFINET IO	No	Optional	Almost unlimited	No	No
PROFINET IRT	No	Optional	Almost unlimited	Yes	Yes
SERCOS III	No	Yes	511	Yes	Yes
SERCOS interface	No	No	254	Yes	Yes
Foundation Fieldbus H1	Yes	No	240	Yes	No
Foundation Fieldbus HSE	No	Yes	Almost unlimited	Yes	No
RAPIDnet	No	Yes	256	Under Development	Conditional
Fieldbus	Bus power	Cabling redundancy	Max devices	Synchronisation	Sub millisecond cycle

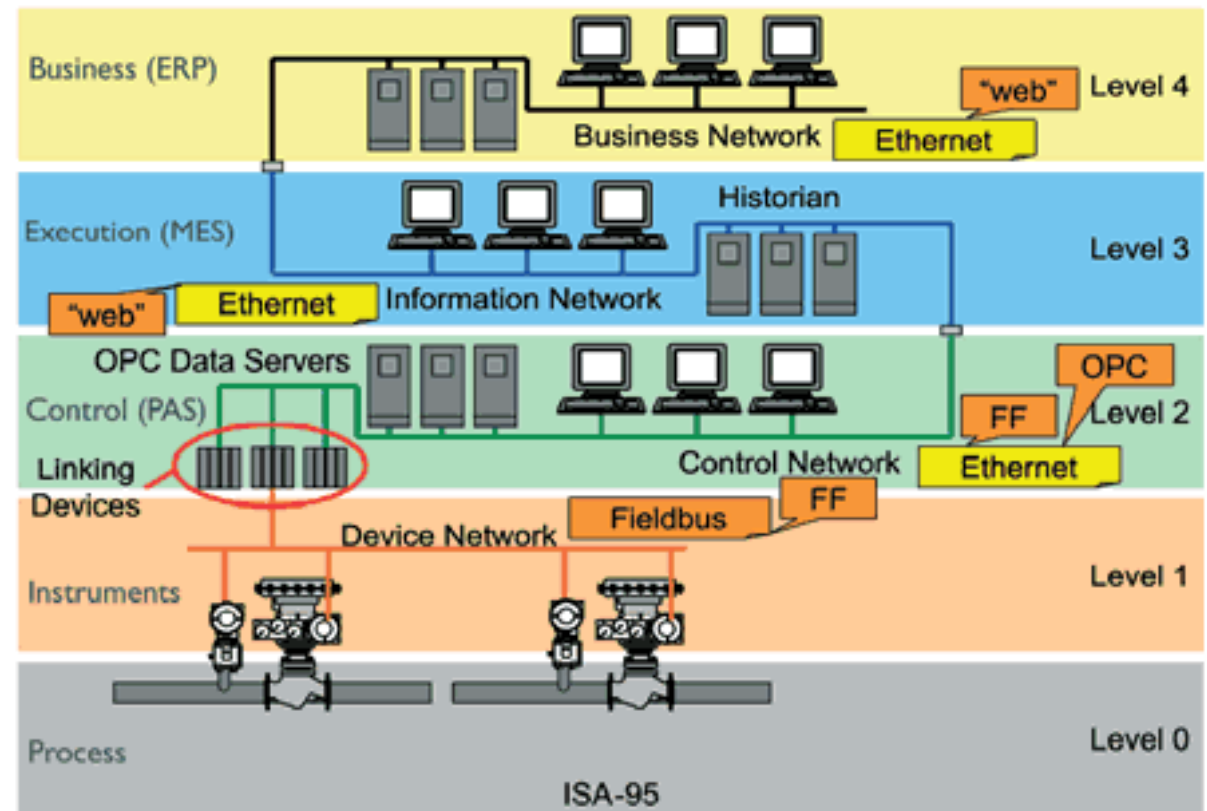
What is it all about ?

- Reliable networking
- Real time behaviour
- Safety Critical ?
- Reliable (sub)systems
- Cheap ?
- + 10 year lifetime
- + 10 year lifetime
- + 10 year lifetime
- tested, documented
- vendor independent ?!



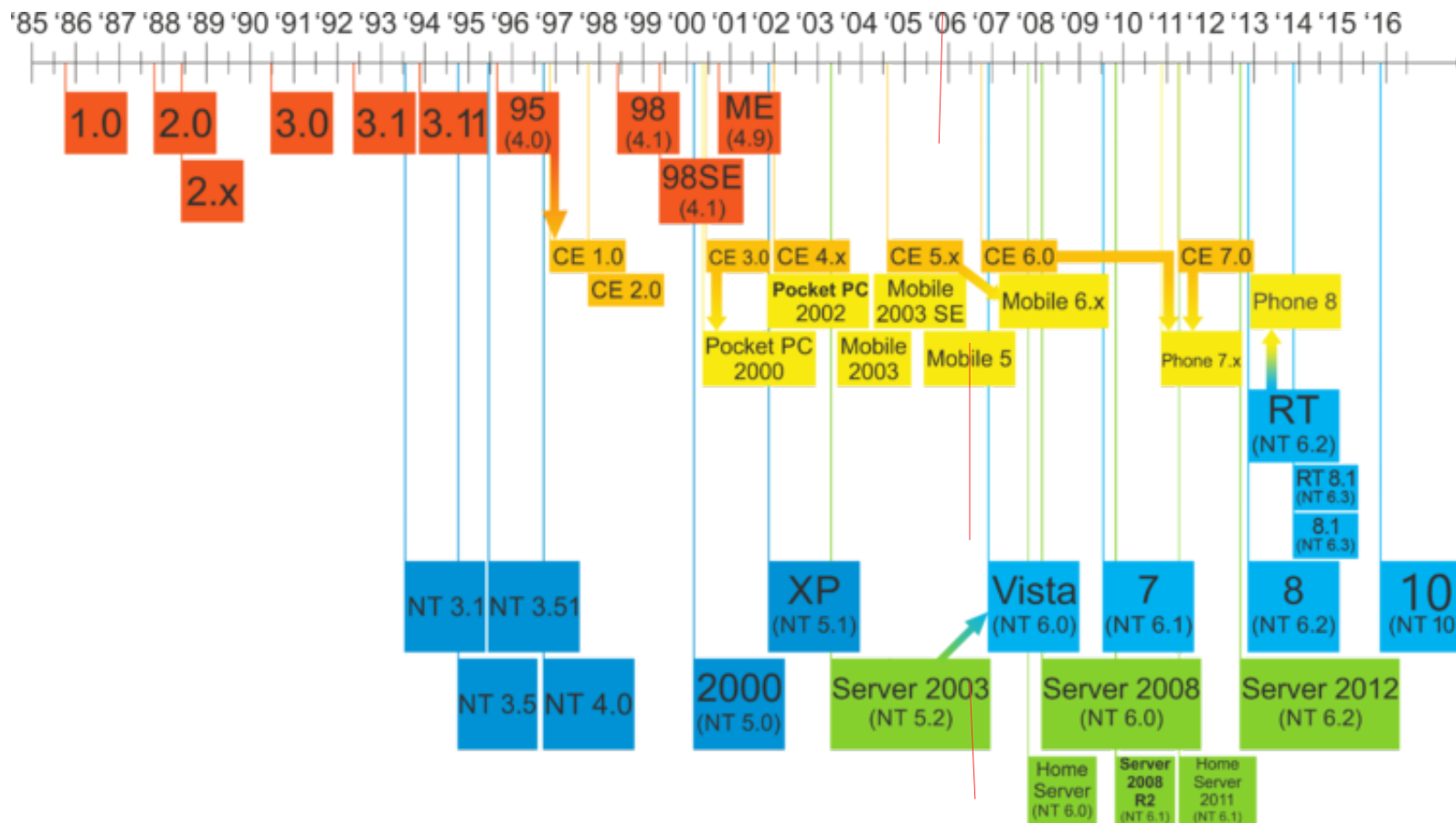
Standards

- Reliable (sub)systems
- Cheap ?
- + 20 year lifetime
- tested, documented
- vendor independent ?!



looking back

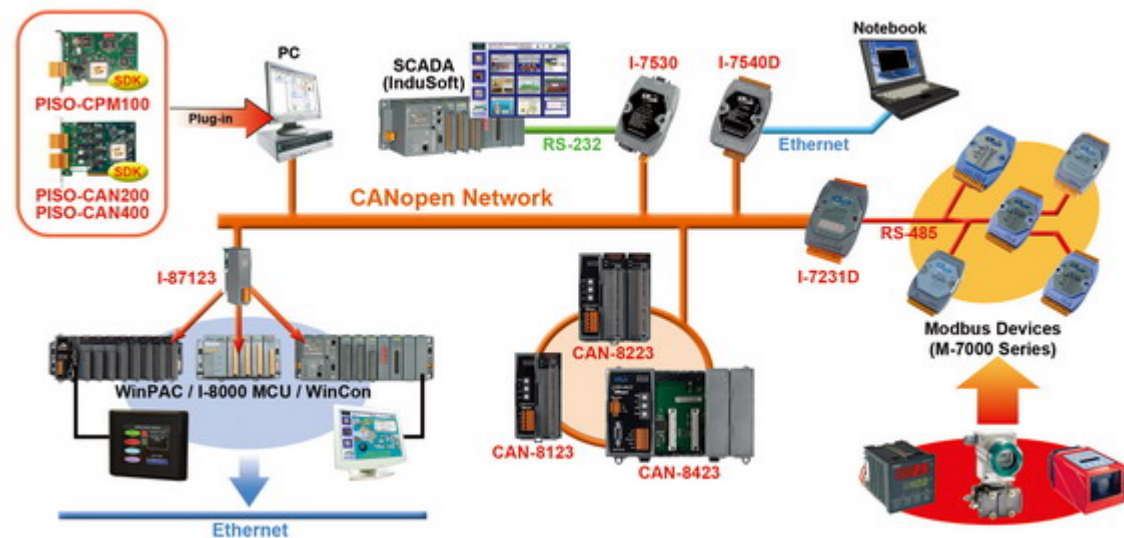
- this year 2016
- In 2006 ...
- and maintenance is still ongoing of you 2006 product



Certification/standards



- Certification
 - maritime
 - avionics
 - military
 - ...
 - ...



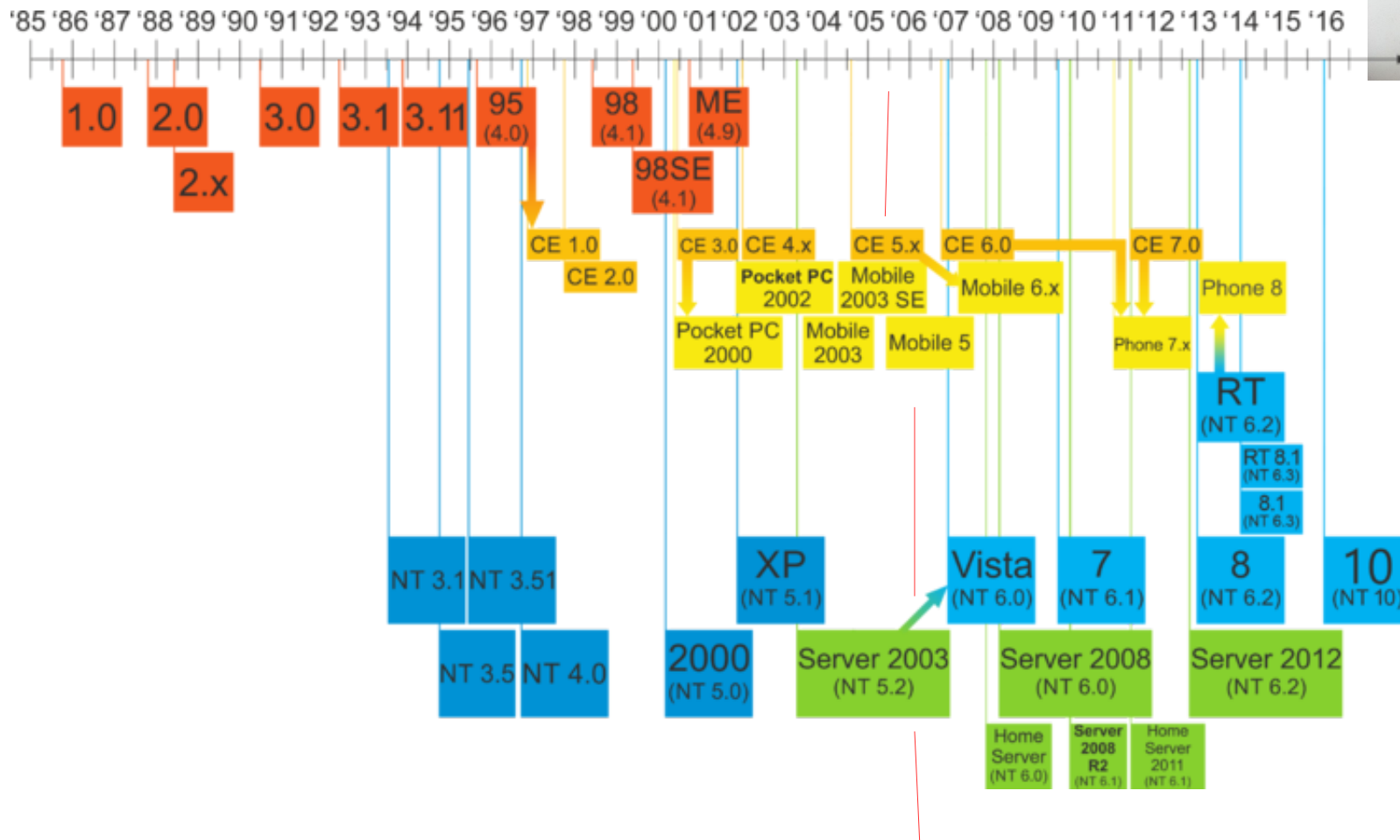
old stuff

- 1968 – start of production !
- new models ..., ~2005
- 1527 built
- ~60 lost (accidents)
- Is still maintained ...



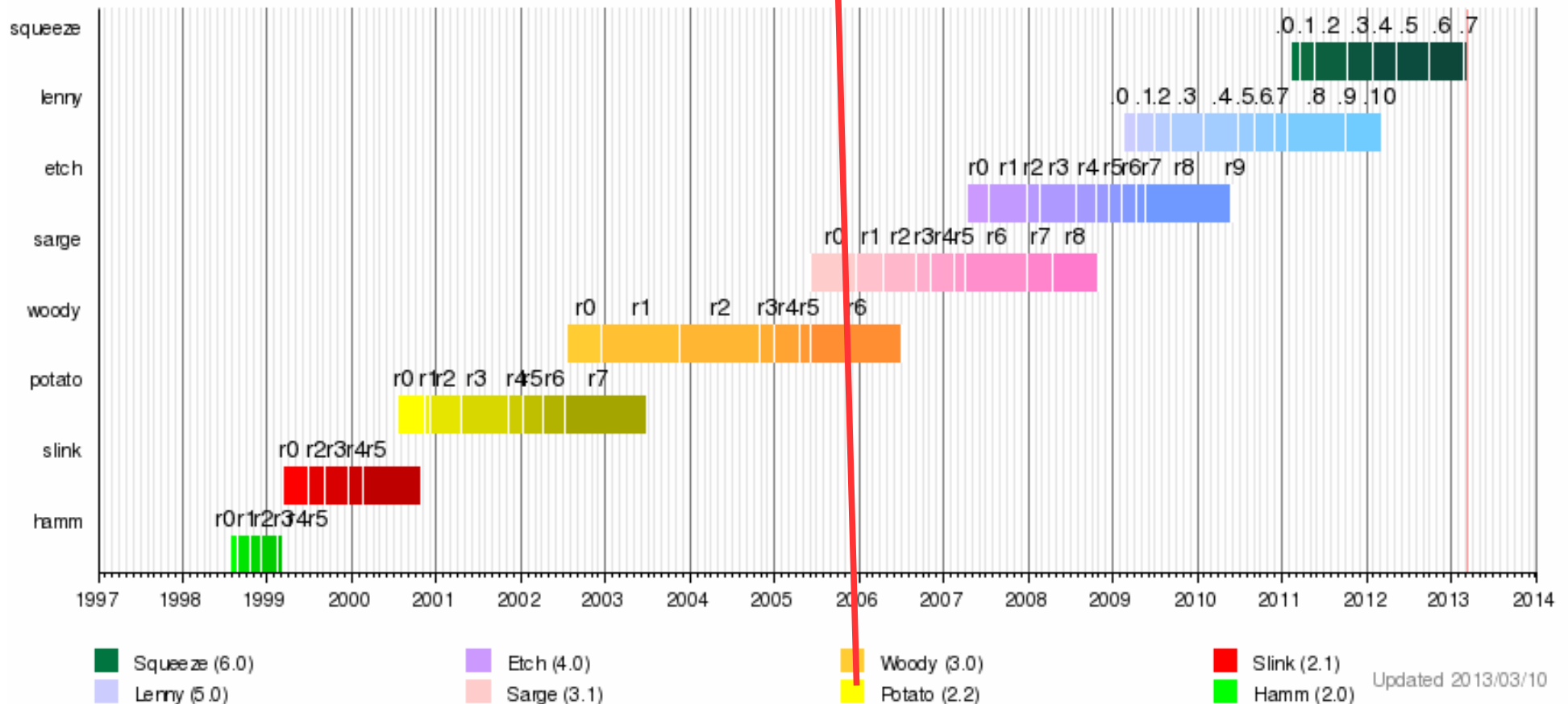
OS discussion - a case

- this year 2016
- In 2006 ...
- and maintenance is still ongoing



stretch (coming)
 jessie (stable 2015->)
 wheezy (obsolete stable 2013->)

Debian GNU/Linux release timeline



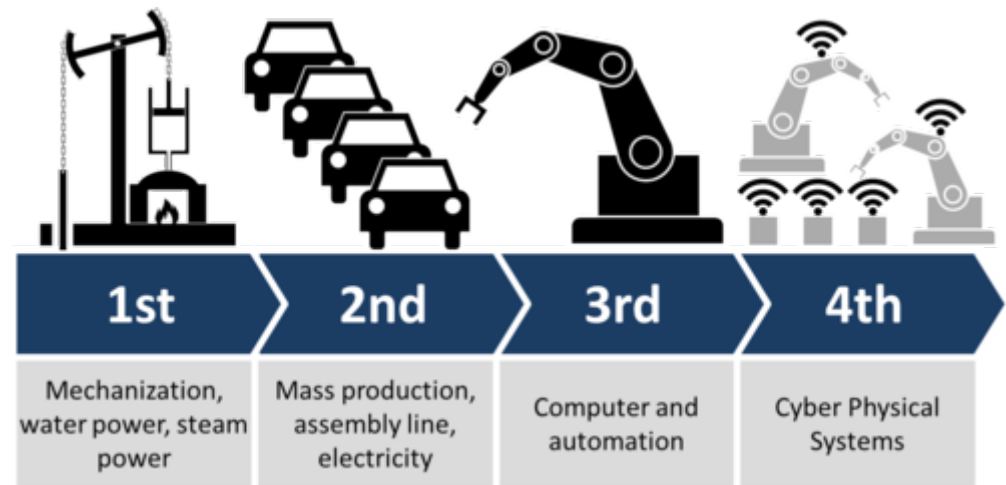
SIL – Safety Integrity Level

- Complicated area ...
- IEC 61508
- A small(!) example
 - Low Demand – max freq 1/year
 - Continuous operation 24/7

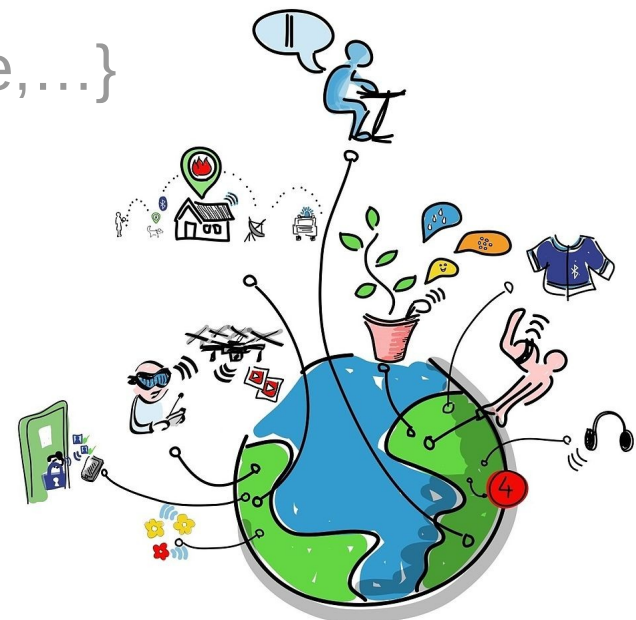
SIL	Low demand mode: average probability of failure on demand	High demand or continuous mode: probability of dangerous failure per hour
1	$\geq 10^{-2}$ to $< 10^{-1}$	$\geq 10^{-6}$ to $< 10^{-5}$
2	$\geq 10^{-3}$ to $< 10^{-2}$	$\geq 10^{-7}$ to $< 10^{-6}$
3	$\geq 10^{-4}$ to $< 10^{-3}$	$\geq 10^{-8}$ to $< 10^{-7}$ (1 dangerous failure in 1140 years)
4	$\geq 10^{-5}$ to $< 10^{-4}$	$\geq 10^{-9}$ to $< 10^{-8}$

IOT :-()

- Standardisation - hmm
- Security - hmm
- Impact on Society - hmm
- Maintenance - hmm



- Can I get my 10 year old {cellphone, stove, ...} upgraded to achieve
 - compatibility
 - safety and security
 - new features ...
 - standardisation.



Basic concept

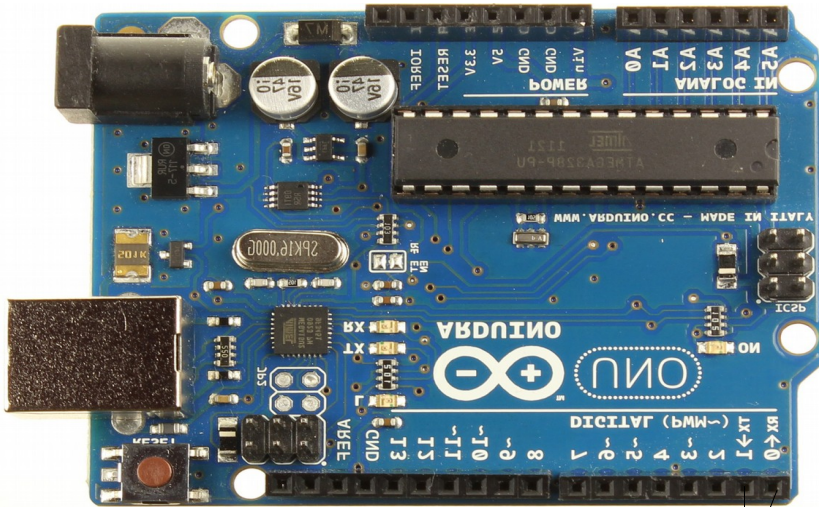


- Self study today
 - slides
 - literature
 - concentration

and ...

BYOD + your effort to

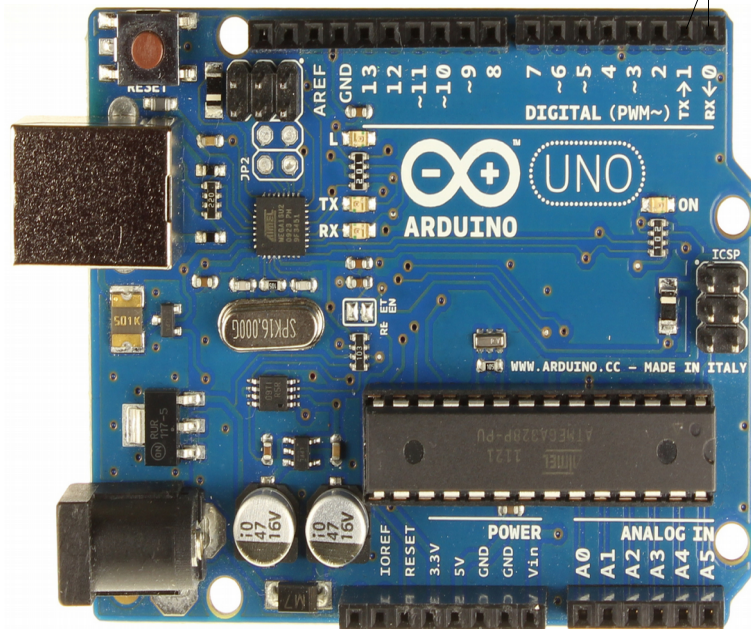
Developpe your own FB :-)



tx-0 -> rx-1

tx-1 -> rx-0

maybe expand ring



the Ring circus

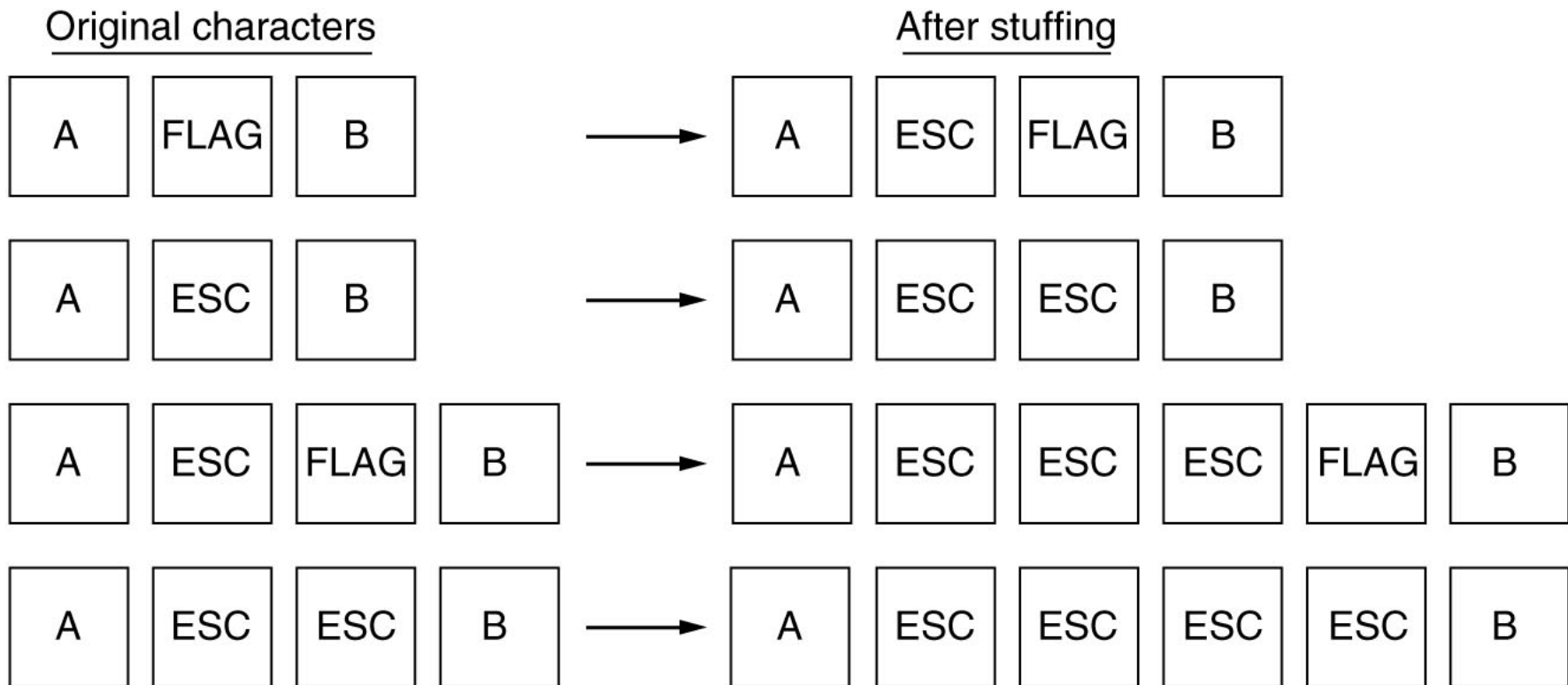
- We need to get a package to circulate
 - from 0 to 1
 - from 1 to 2
 - from 2 to 3
 - from 3 to 0
 - remove it ?
 - change info in package ?
 - or ?

- Constraints
 - serial interface
 - only visible ascii is allowed in packet ('a','b','A',... '0','1',...)
- Byte stuffing (Ex.1 Srch for hdlc and/or byte stuffing)
- Asynchronous channel (rs232)
- Definition of frame format

Stuffing for the stuffed



(a)



(b)

- What to move around ?
 - fixed size data (so it fits into Arduino RX Buffer)
 - -> your choice ?
 - int (as text)

```
#if (RAMEND < 1000)
  #define SERIAL_BUFFER_SIZE 16
#else
  #define SERIAL_BUFFER_SIZE 64
#endif
```

```
struct ring_buffer
{
  unsigned char buffer[SERIAL_BUFFER_SIZE];
  volatile unsigned int head;
  volatile unsigned int tail;
};
```

- Distributed Memory ?
 - aka broadcast - all shall read same package
 - Src remove it from the ring
- Specific Node Adressing ?
 - src, dst
- Integrity
 - checksum :-) crc-16/32

From application point of view



- Do not care about fieldbus ?
 - distr/shared memory
- Care about ...
 - Specific transfer info to/from node
 - `send(dst, "var1"); ...`

- **Exercise today**
-
- define simple frame format
 - is byte stuffing needed (depending on your design)
- At data link layer TX
- RX (a little bit more nasty bq you need to sync to frame delimiter)

- design API for application
 - `wr(int varNumber, int value)`
 - `rd (int varNumber, int *value)`

- TIMING ?
 - how to measure
- Errors
 - how to test (hint inject your own errors)