Hacía una mejor instrumentación (con Python y Lantz)

Hernán E. Grecco
hgrecco@df.uba.ar

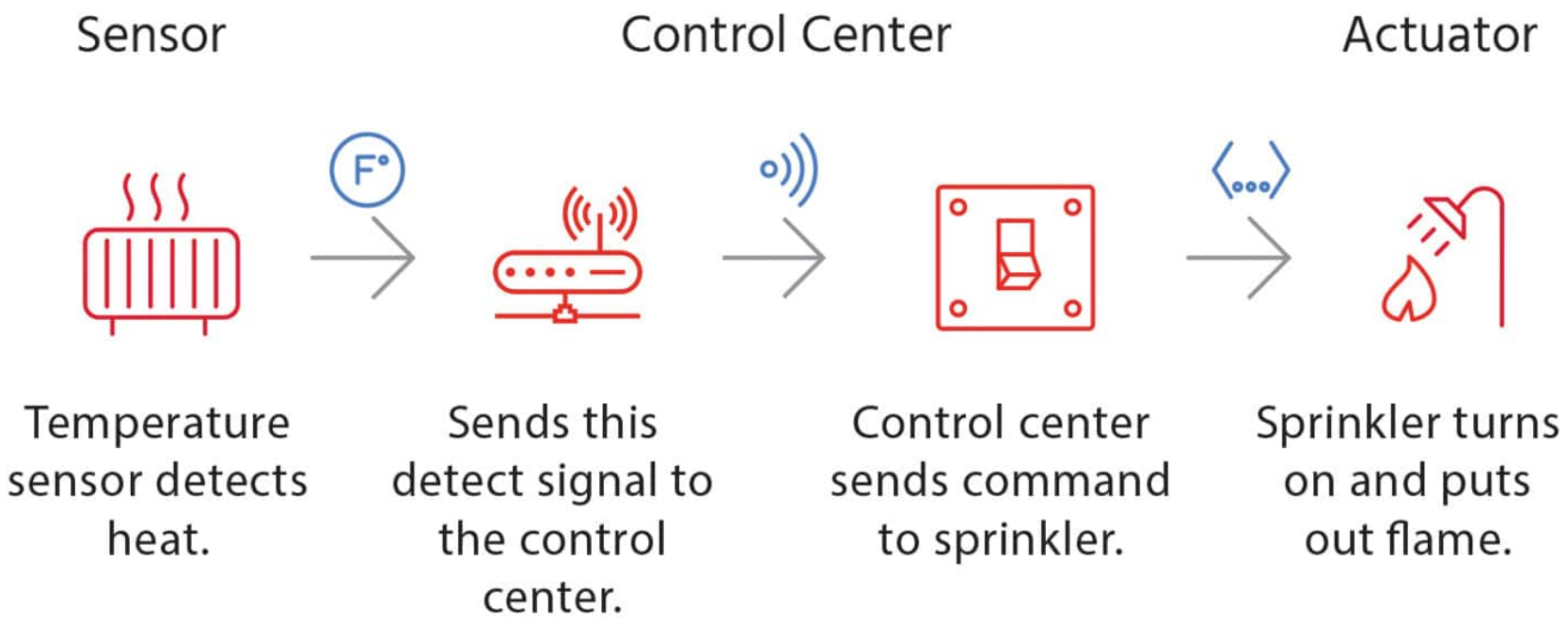
Mayo de 2018

Maximiliano Lantz

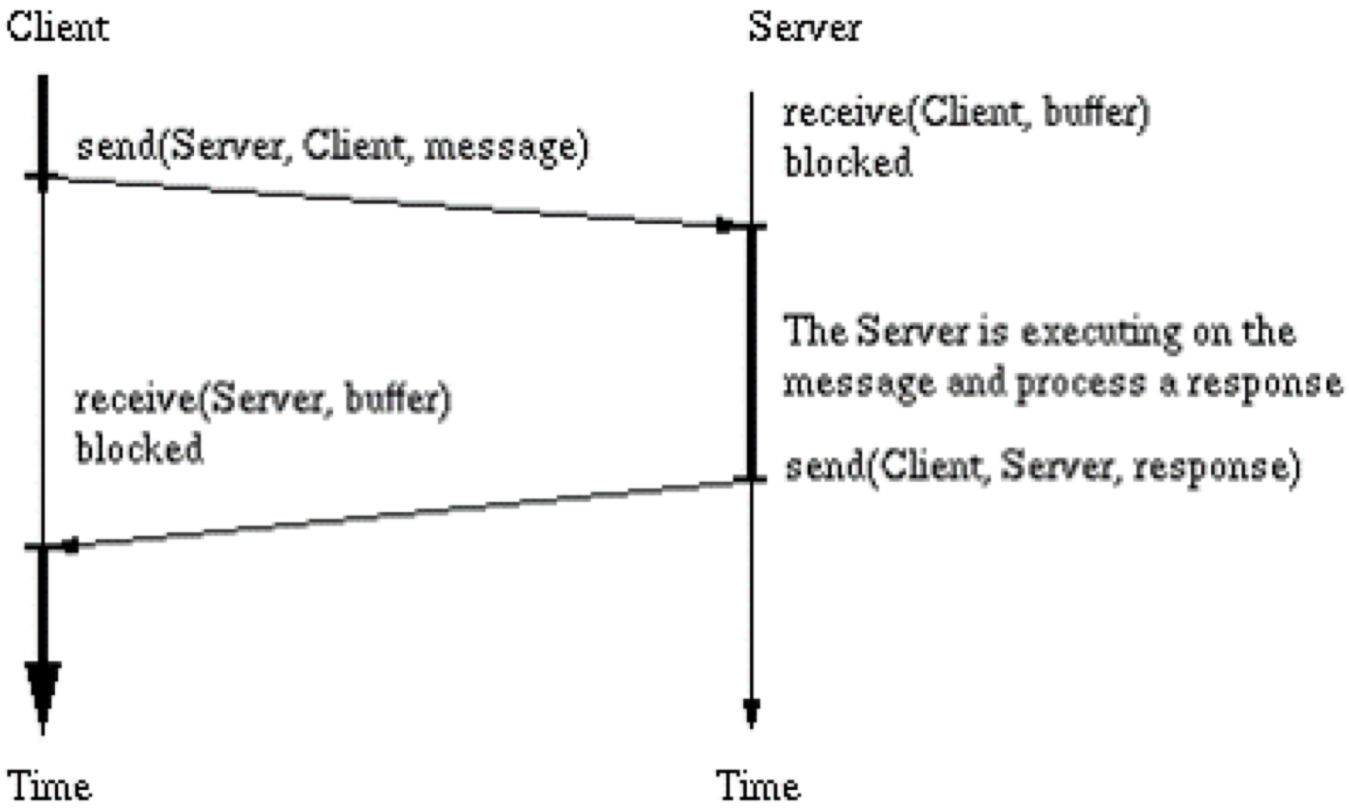
Aula 12 del Pabellón II, Ciudad Universitaria

Observación e Intervención

Sensores y Actuadores



Hablando con Instrumentos

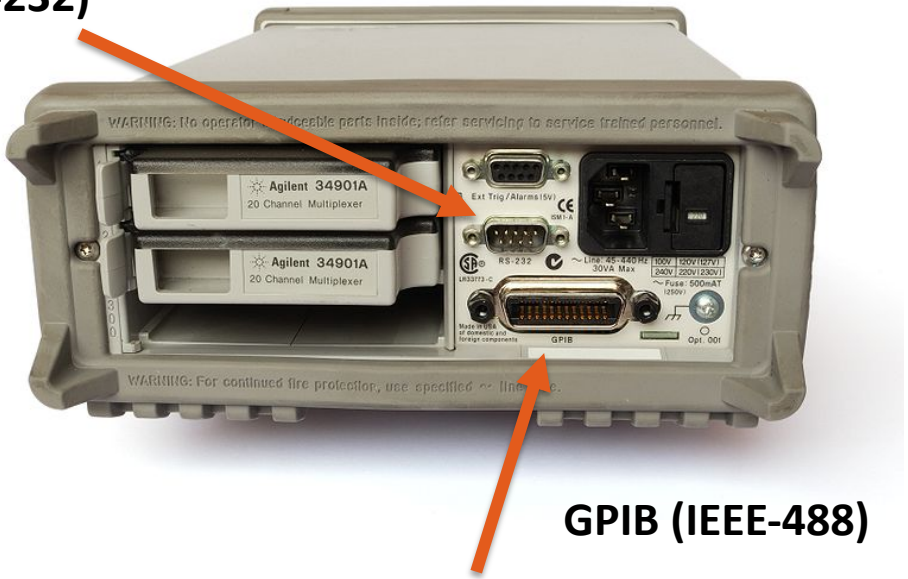


Hablando con Instrumentos

Ethernet

USB

Serie (RS-232)

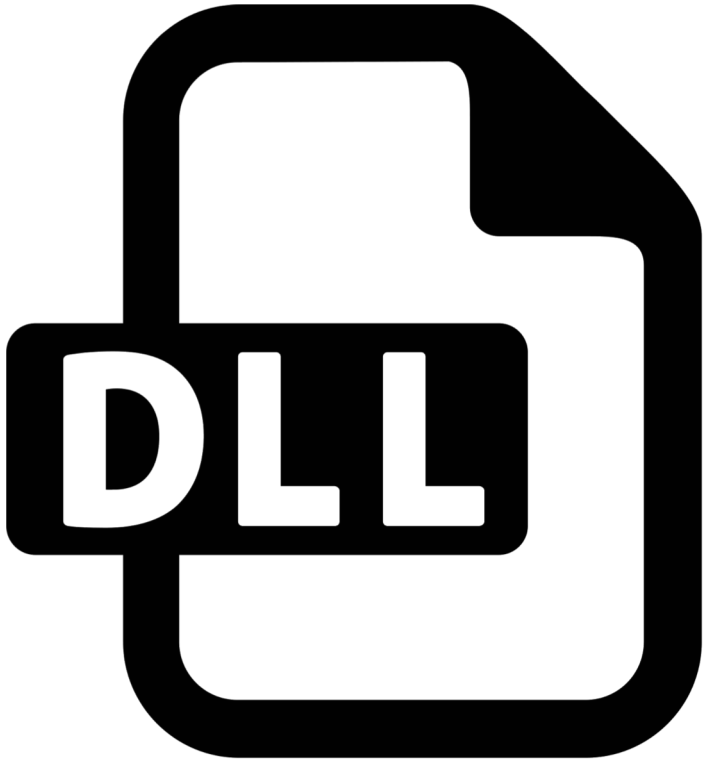


GPIB (IEEE-488)

Hablando con Instrumentos

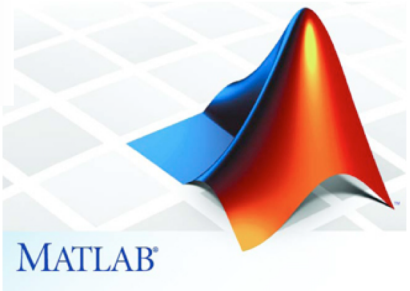
Interface	GPIB	LAN	USB	RS232
				
Bandwidth	1.8Mbit/s (488.1) 	12.5 Mbit/s 125 Mbits/s (GB LAN) 	60 Mbit/s (Hi-Speed) 120 Mbit/s (USB 3.0) 	28.8 kB/s 
Latency (informative) *Note1	300 µs 	250 µs 	1000 µs (USB) 125 µs (Hi-Speed) 	~100 ms 

Hablando con Instrumentos



Opciones para instrumentación y control

Domain Specific Languages (DSL)



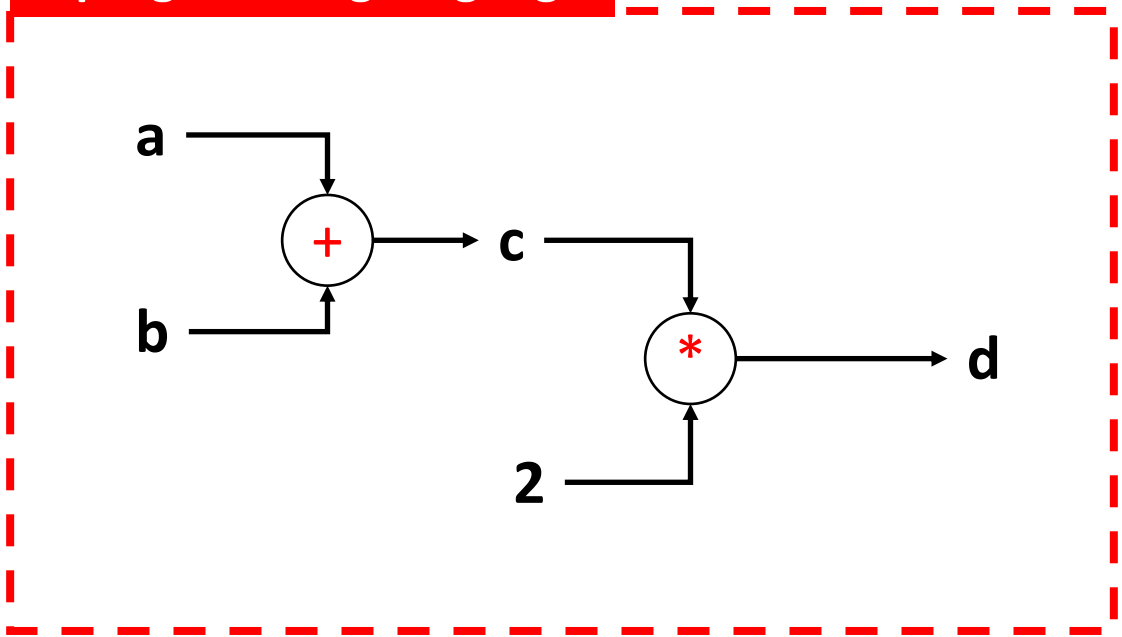
General Purpose Languages (GPL)
+ add ons

Measurement Studio

Los DSL tienen un propósito específico (y a veces nos joden)

LABoratory **V**irtual **I**nstrumentation **E**ngineering **W**orkbench

G programming language

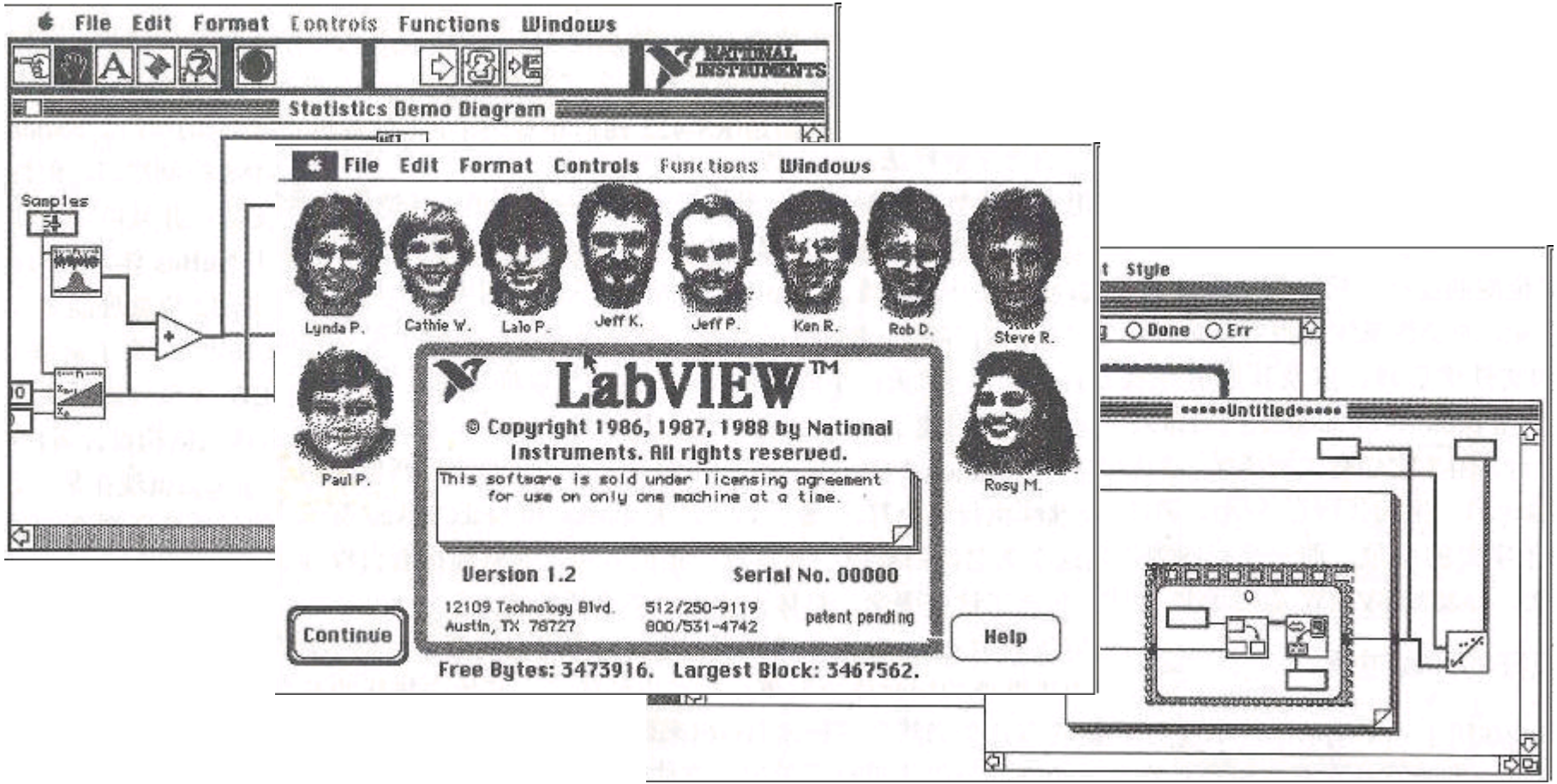


Imperative Programming

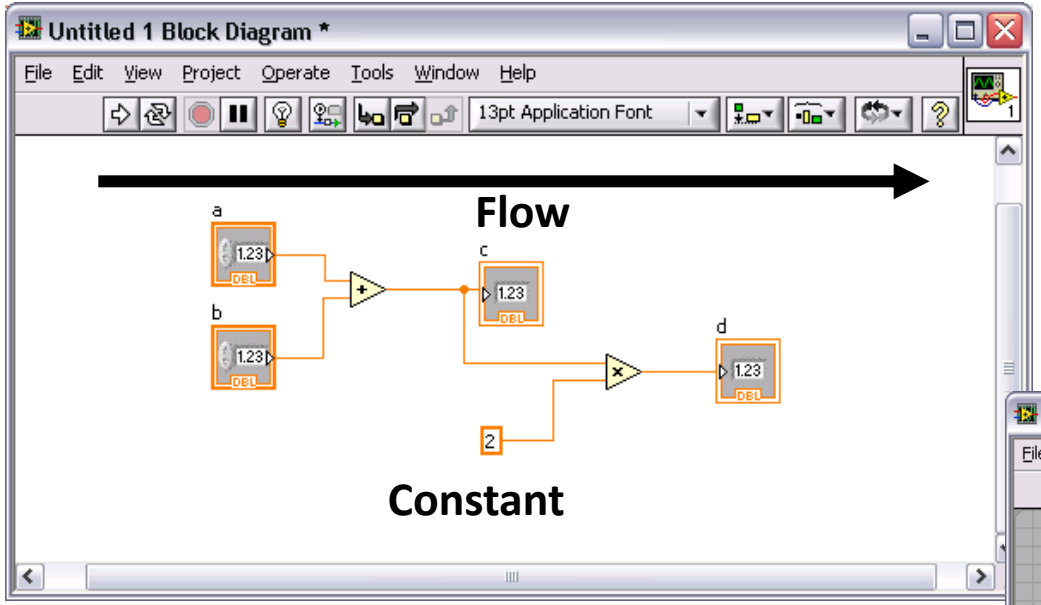
c = a + b

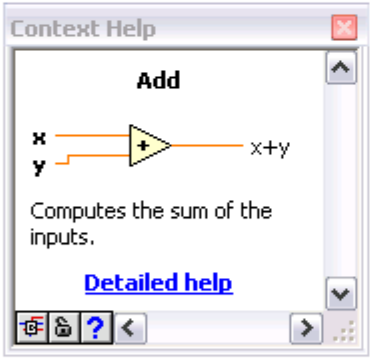
d = 2*c

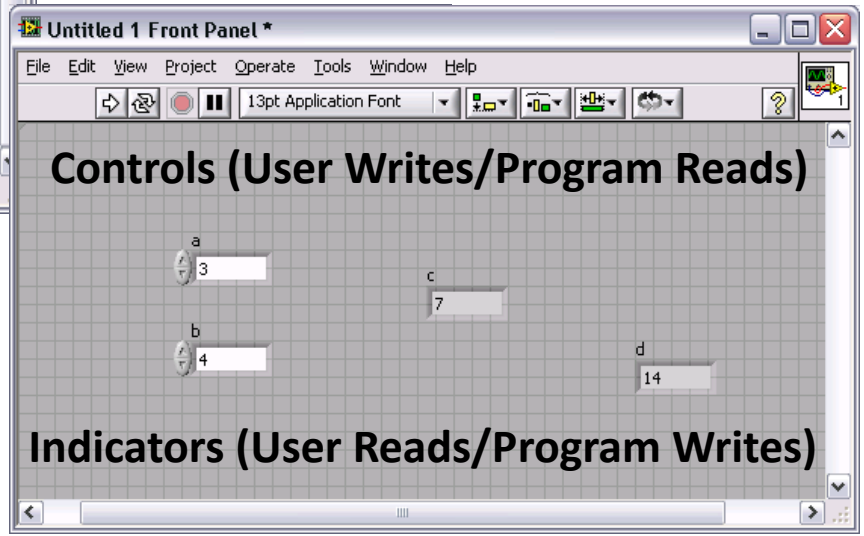
Los DSL tienen un propósito específico (y a veces nos joden)



Los DSL tienen un propósito específico (y a veces nos joden)



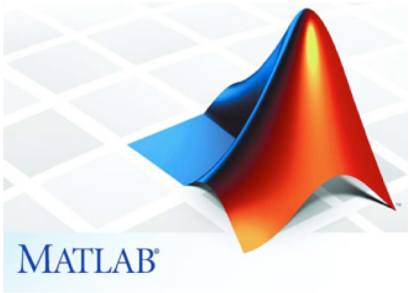




vi = Virtual Instrument

Cuando duele: String manipulation, access to databases, User interfaces, Object oriented programming, source control, documentation, testing …

Los DSL tienen un propósito específico (y a veces nos joden)



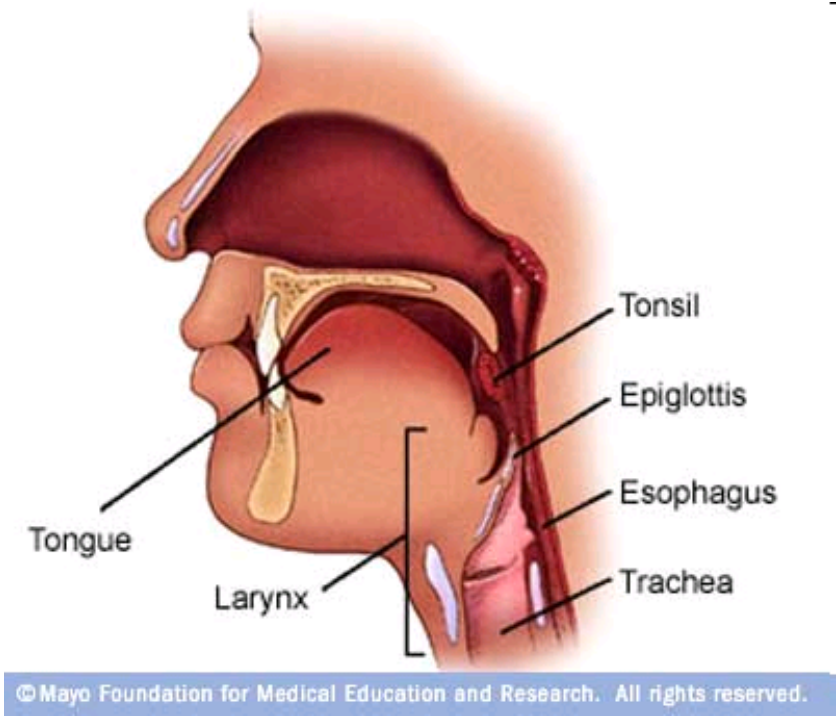
MATrix **LAB**oratory

```
>> A = pascal(3);
>> u = [3; 1; 4];
>> x = A\u
x =
     10
    -12
      5
```

```
>> g = gpib('ni',0,1);
>> fopen(g)
>> set(g,'EOSMode','read&write’)
>> set(g,'EOSCharCode','LF')
>> fprintf(g,'Volt 3')
>> fclose(g)
>> delete(g)
>> clear g
```

Cuando duele: String manipulation, access to databases, User interfaces, No namespaces => Weird names functions, Object oriented programming, documentation, testing …

Los DSL se atragantan con la vida real



5051 muertes en USA en 2017

Los GPL + libs son pagos o complejos

ANSI C + libraries to control instruments and common controls (plots, etc)
Memory management, bookkeeping

Measurement Studio

.NET libraries (VB, C#, etc) to control instruments and common controls (plots, etc)

5051 muertes en USA en 2017

Yo quiero hacer instrumentación con un Lenguaje

- Propósito general.
- Prototipado rápido
- 

- Extensible con C/C++/Fortran/Rust
- Gratis and Libre
- Typos dinámicos pero estrictos. Con librerías científicas.

Se puede hacer instrumentación con un Python

```
import serial

inst = serial.Serial('COM1')
inst.write('AMPLITUDE?\n')
amplitude = float(inst.recv())
print(amplitude)
inst.write('AMPLITUDE 5.00\n')
inst.write('SCAN 10.00 100.00\n')
```

Se puede hacer instrumentación con un Python

```python
import serial

class SignalGenerator(object):

    def __init__(self, port):
        self.serial = serial.Serial(port)

    @property
    def amplitude(self):
        self.serial.write('AMPLITUDE?\n')
        return float(self.serial.recv())

    @amplitude.setter
    def amplitude(self, value):
        self.serial.write('AMPLITUDE {}\n'.format(value))

    def frequency_scan(self, first, last):
        self.serial.write('SCAN {} {}\n'.format(first, last))


inst = SignalGenerator('COM1')
amplitude = inst.amplitude
print(amplitude)
inst.amplitude = 5
inst.frequency_scan(10, 100)
```

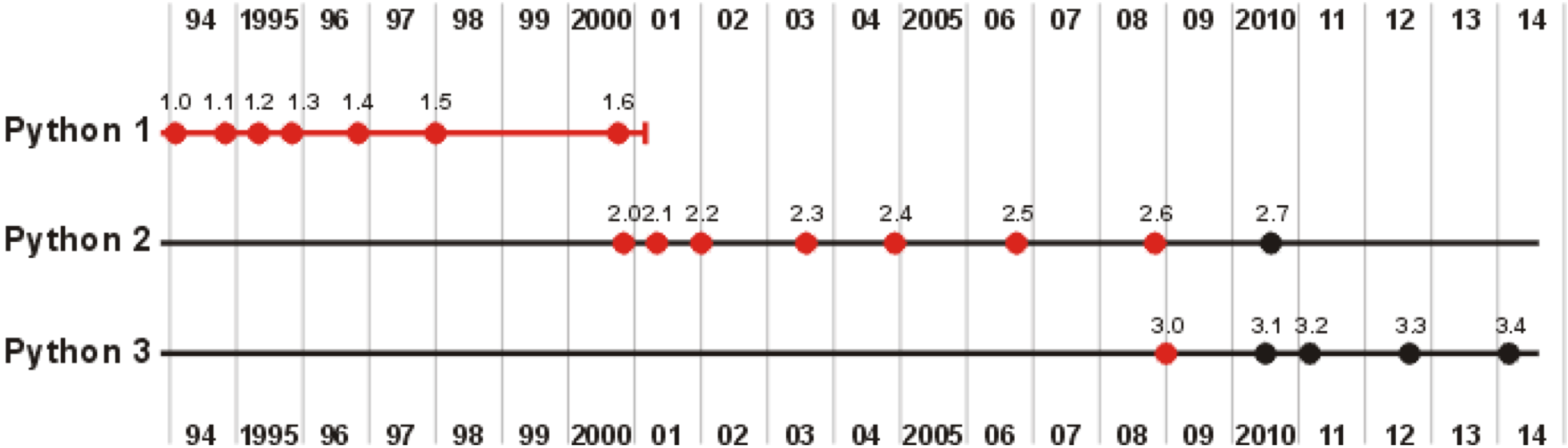
Se puede hacer instrumentación con un Python

```
@amplitude.setter
def amplitude(self, value):
    if value == self._last_amplitude:
        if DEBUG:
            log('No need to change the amplitude')
            return

    if DEBUG:
        log('Changing the amplitude to {}'.format(value))

    self.serial.write('AMPLITUDE {}\n'.format(value))

    if DEBUG:
        log('Changed the amplitude to {}'.format(value))
```


Lantz
python
powered

- Lantz es un paquete, Python es el lenguaje.
- Proveer soluciones a problemas comunes.
- Dejar los casos particulares/patológicos para aquellos que los tengan.
- Ayudar a “Hacer las Cosas Bien”
- Simplificar la identificación de error
- Favorecer que el código se comparta

Python: Historia



Implementaciones: CPython, Jython, IronPython, PyPy, …

Python: Batteries Included

Ecosistema de Python: Repositorio de Paquetes (PyPi)

Instaladores: pip (Pip Installs Packages)

No usar: easy_install

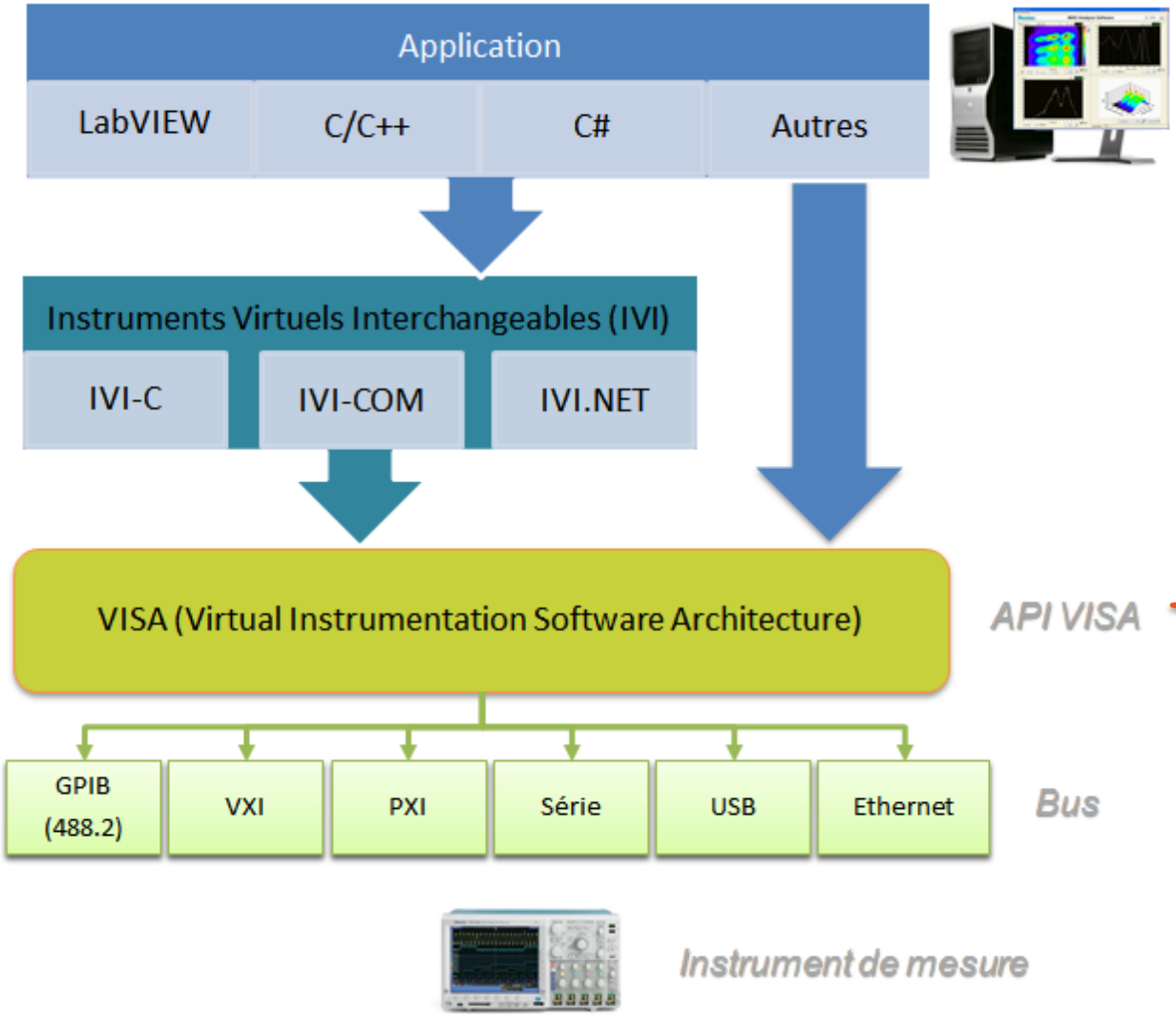
Ecosistema de Python: Distribuciones

Instaladores: conda

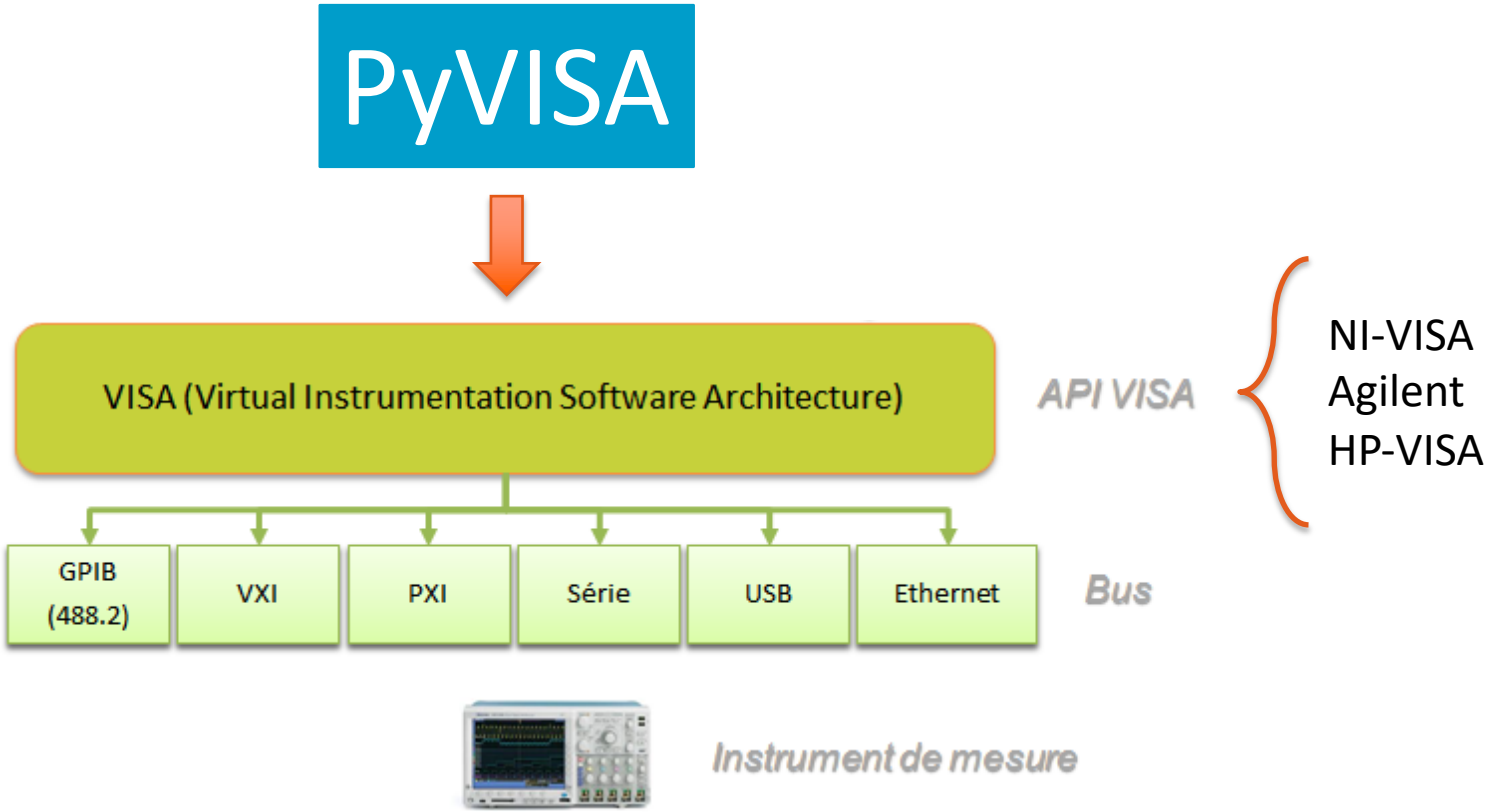
Ecosistema de Python para instrumentación

- DLL: built-in
- Ethernet: built-in
- Serial: PySerial
- USB: PyUSB
- GPIB: linux-gpib

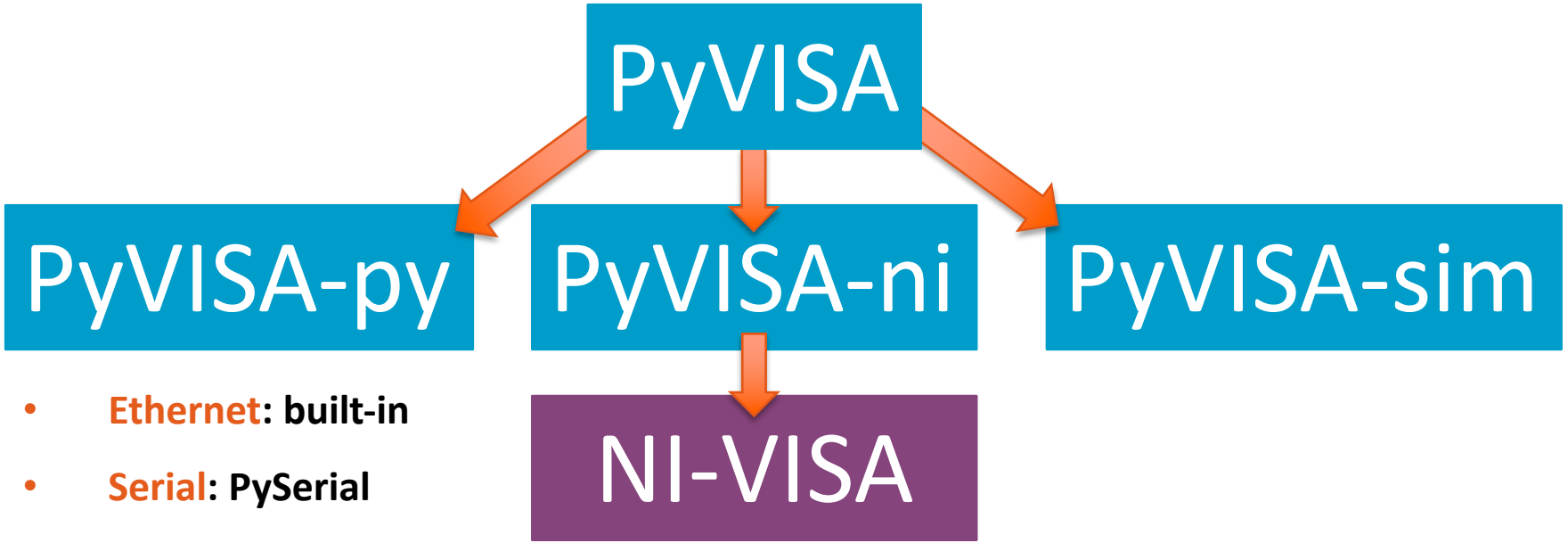
Mensajes que van y vienen


Application
LabVIEW
C/C++
C#
Autres
Instruments Virtuels Interchangeables (IVI)
IVI-C
IVI-COM
IVI.NET
VISA (Virtual Instrumentation Software Architecture)
API VISA
NI-VISA
Agilent
HP-VISA
GPIB (488.2)
VXI
PXI
Série
USB
Ethernet
Bus
Instrument de mesure

Ecosistema de Python para instrumentación

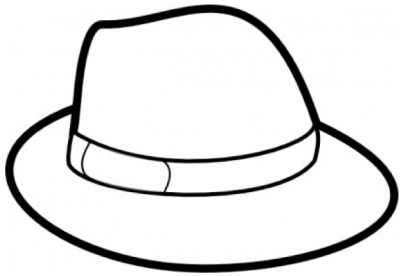


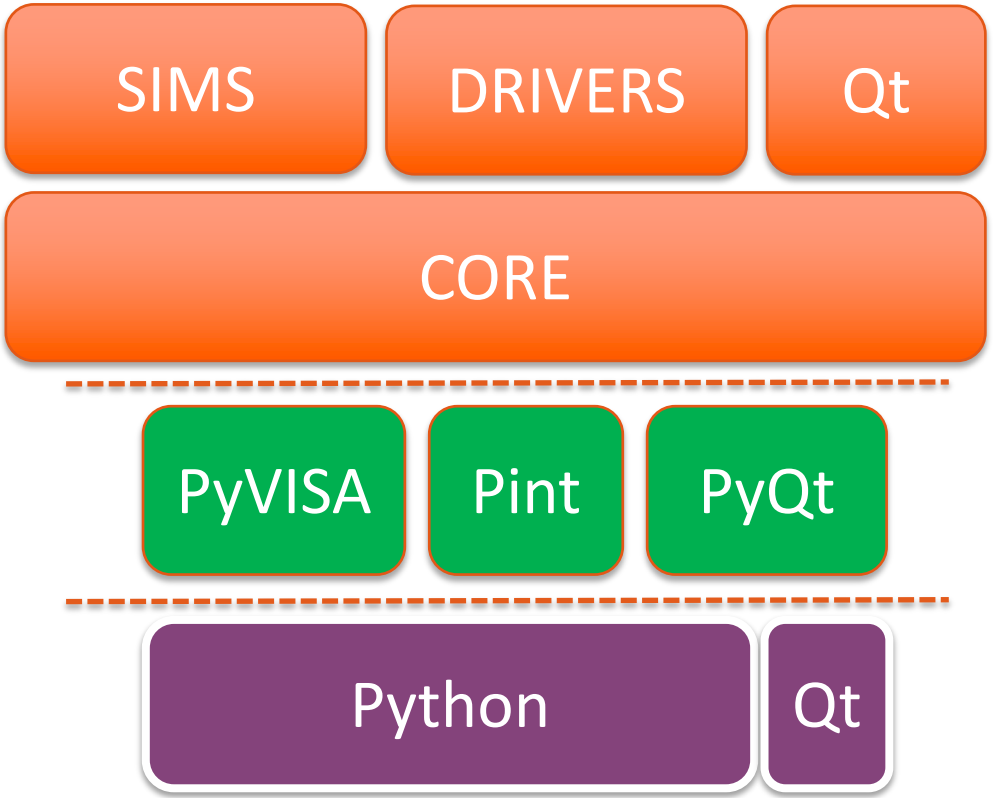
Ecosistema de Python para instrumentación



- **Ethernet**: built-in
- **Serial**: PySerial
- **USB**: PyUSB
- **GPIB**: linux-gpib

Ecosistema de Python para instrumentación





Ecosistema de Python para instrumentación

https://github.com/lantzproject