cosylab

CONTROL SYSTEM LABORATORY

CAN THE ACCELERATOR CONTROL SYSTEM BE BOUGHT FROM INDUSTRY?

Mark Pleško

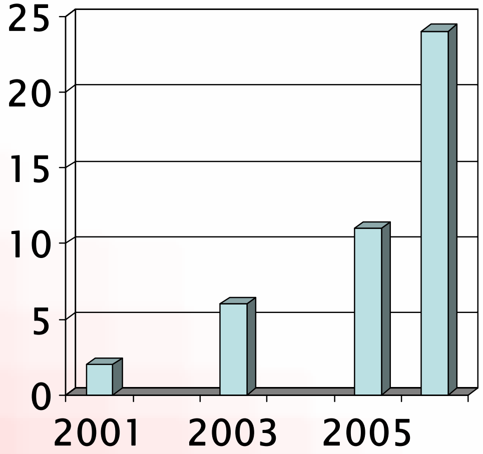
mark.plesko@cosylab.com

EPAC 06, Edinburgh

How We Started the Company

- **"Spin-off" from the Josef Stefan Institute**
  - **1996**: M.Pleško is lucky to get a few top students
  - **1999**: Soft- and Hardware for the ANKA Light Source
  - **2001**: can't keep graduates - company start-up
    - Specialized in accelerator and beam line control systems
    - Continue to hire excellent people
- **Growth from own means:**

Cosylab team	**2001**	**2003**	**2005**
Full time	2	6	13
Nearly full time	6	6	15
Projects	4	15	30
Customers	4	12	25

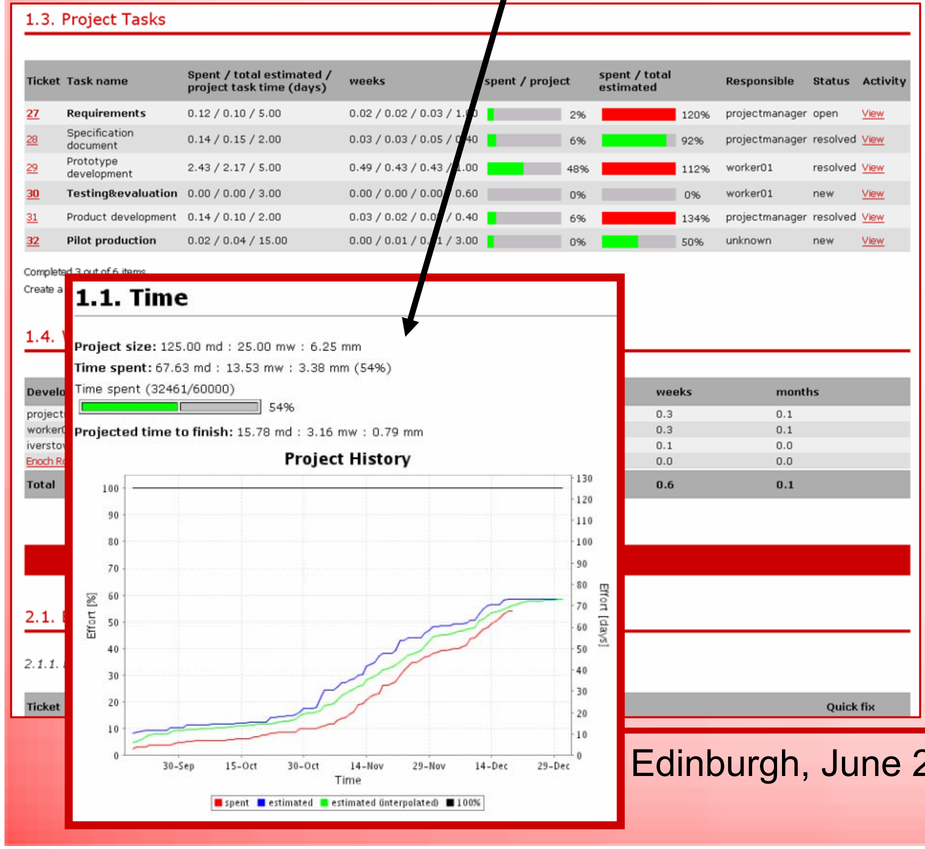


Not How, But Who Does It

- **A real "Spin-off"**
  - 1 researcher and 5 graduates must earn their living
  - Research competence and business culture
- **Getting the best people**
  - Recruit the most talented undergraduate students
  - Add culture and loyalty (also through shares)
  - Money is less important than one thinks
    - it's a negative motivator
    - Must find positive motivators, too!
  - Now over 50 students in the pipeline, from simple exercises to production work

Own Project Management/Reporting Software

- Web/email ticketing system
- Measure and analyse time
- “on-click” project reports



1.3. Project Tasks

Ticket	Task name	Spent / total estimated / project task time (days)	weeks	spent / project		spent / total estimated		Responsible	Status	Activity
27	**Requirements**	0.12 / 0.10 / 5.00	0.02 / 0.02 / 0.03 / 1.00	🟩	2%	🟥	120%	projectmanager	open	View
28	Specification document	0.14 / 0.15 / 2.00	0.03 / 0.03 / 0.05 / 0.40	🟩	6%	🟩	92%	projectmanager	resolved	View
29	Prototype development	2.43 / 2.17 / 5.00	0.49 / 0.43 / 0.43 / 1.00	🟩	48%	🟥	112%	worker01	resolved	View
30	**Testing&evaluation**	0.00 / 0.00 / 3.00	0.00 / 0.00 / 0.00 / 0.60		0%		0%	worker01	new	View
31	Product development	0.14 / 0.10 / 2.00	0.03 / 0.02 / 0.0[illegible] / 0.40	🟩	6%	🟥	134%	projectmanager	resolved	View
32	**Pilot production**	0.02 / 0.04 / 15.00	0.00 / 0.01 / 0.[illegible]1 / 3.00	🟩	0%	🟩	50%	unknown	new	View

Completed 3 out of 6 items

Create a

1.4.

Develo		weeks	months
project		0.3	0.1
worker0		0.3	0.1
iverstov		0.1	0.0
Enoch R		0.0	0.0
Total		**0.6**	**0.1**

2.1.

2.1.1.

Ticket	Quick fix

1.1. Time

Project size: 125.00 md : 25.00 mw : 6.25 mm

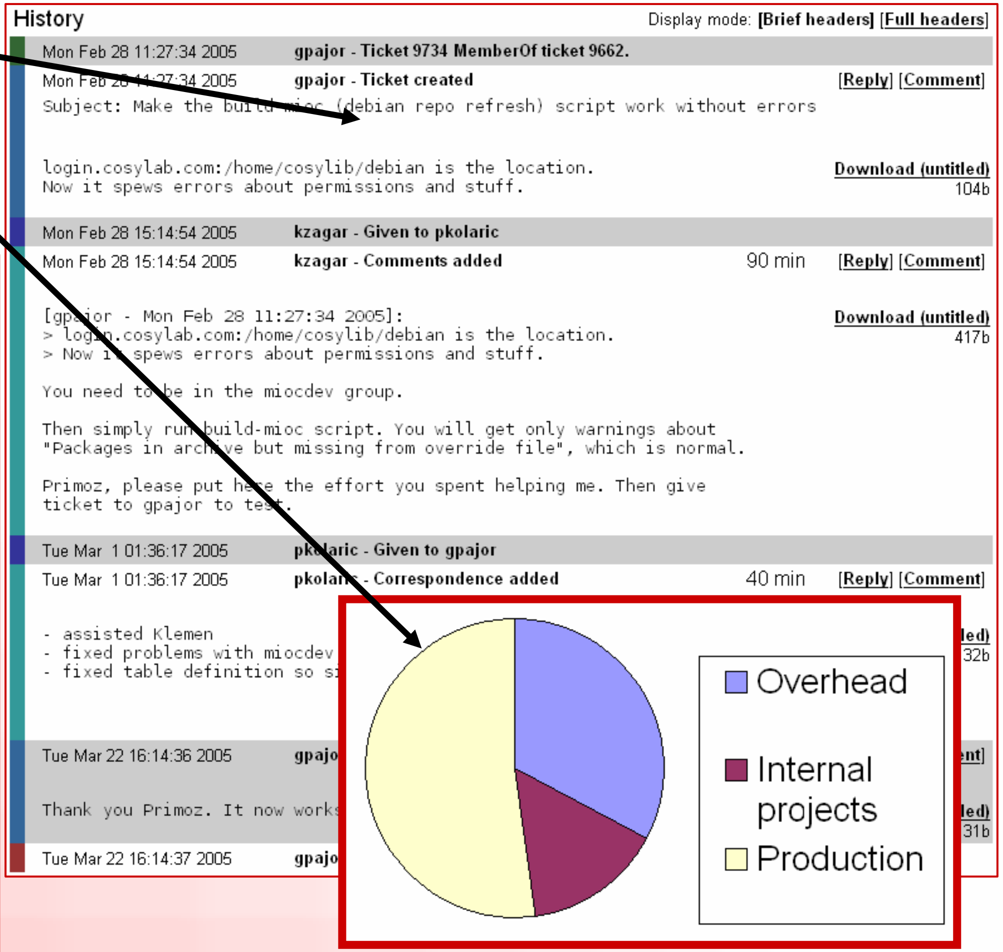
Time spent: 67.63 md : 13.53 mw : 3.38 mm (54%)

Time spent (32461/60000)

54%

Projected time to finish: 15.78 md : 3.16 mw : 0.79 mm

Project History



History

Display mode: **[Brief headers]** [**Full headers**]

Mon Feb 28 11:27:34 2005 **gpajor - Ticket 9734 MemberOf ticket 9662.**

Mon Feb 28 11:27:34 2005 **gpajor - Ticket created** [**Reply**] [**Comment**]

```
Subject: Make the build-mioc (debian repo refresh) script work without errors


login.cosylab.com:/home/cosylib/debian is the location.
Now it spews errors about permissions and stuff.
```

Download (untitled) 104b

Mon Feb 28 15:14:54 2005 **kzagar - Given to pkolaric**

Mon Feb 28 15:14:54 2005 **kzagar - Comments added** 90 min [**Reply**] [**Comment**]

```
[gpajor - Mon Feb 28 11:27:34 2005]:
> login.cosylab.com:/home/cosylib/debian is the location.
> Now it spews errors about permissions and stuff.

You need to be in the miocdev group.

Then simply run build-mioc script. You will get only warnings about
"Packages in archive but missing from override file", which is normal.

Primoz, please put here the effort you spent helping me. Then give
ticket to gpajor to test.
```

Download (untitled) 417b

Tue Mar 1 01:36:17 2005 **pkolaric - Given to gpajor**

Tue Mar 1 01:36:17 2005 **pkolaric - Correspondence added** 40 min [**Reply**] [**Comment**]

```
- assisted Klemen
- fixed problems with miocdev
- fixed table definition so si
```

led) 32b

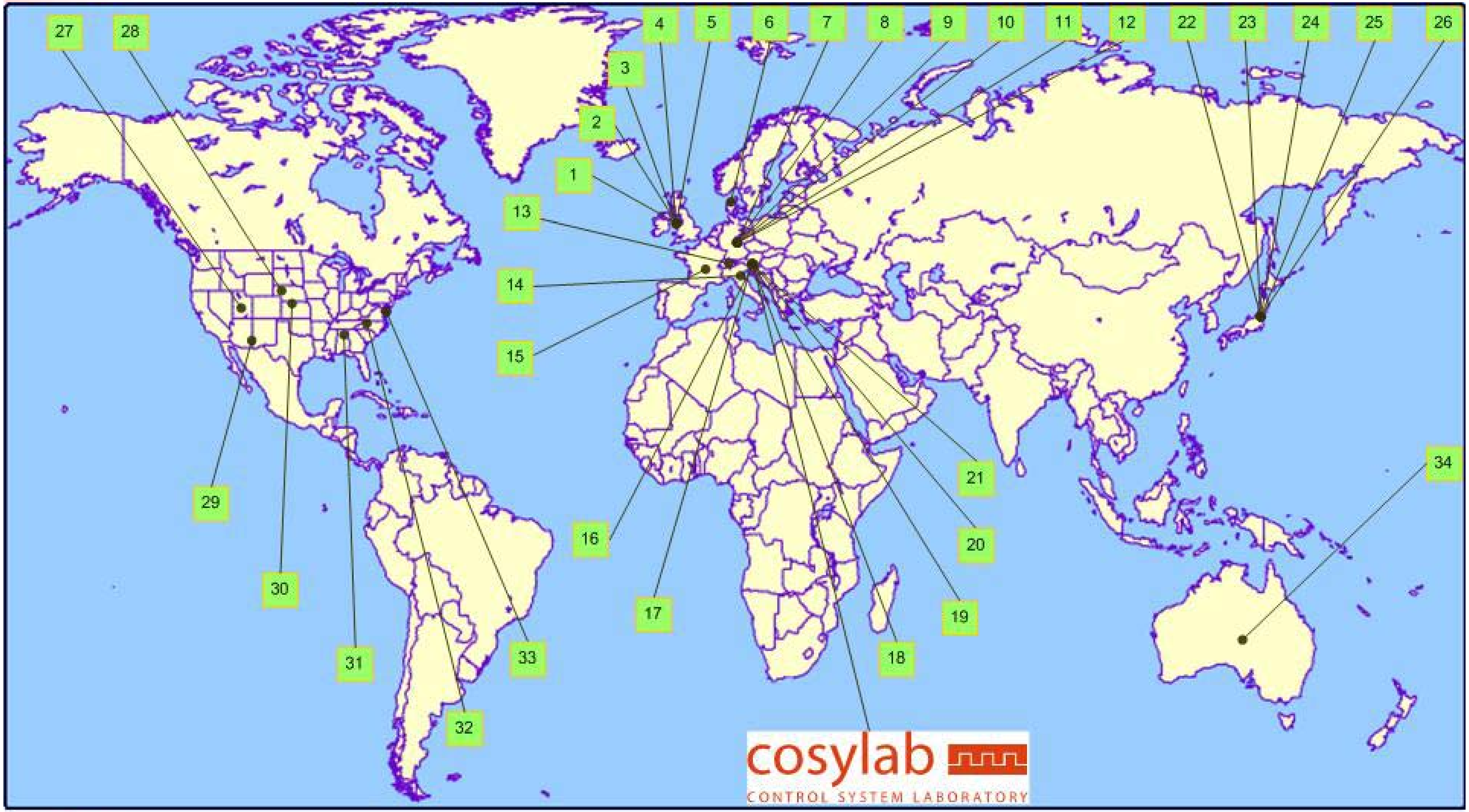
Tue Mar 22 16:14:36 2005 **gpajo** **ent]**

```
Thank you Primoz. It now works
```

led) 31b

Tue Mar 22 16:14:37 2005 **gpajo**

Cosylab Customer Base



LEGEND:

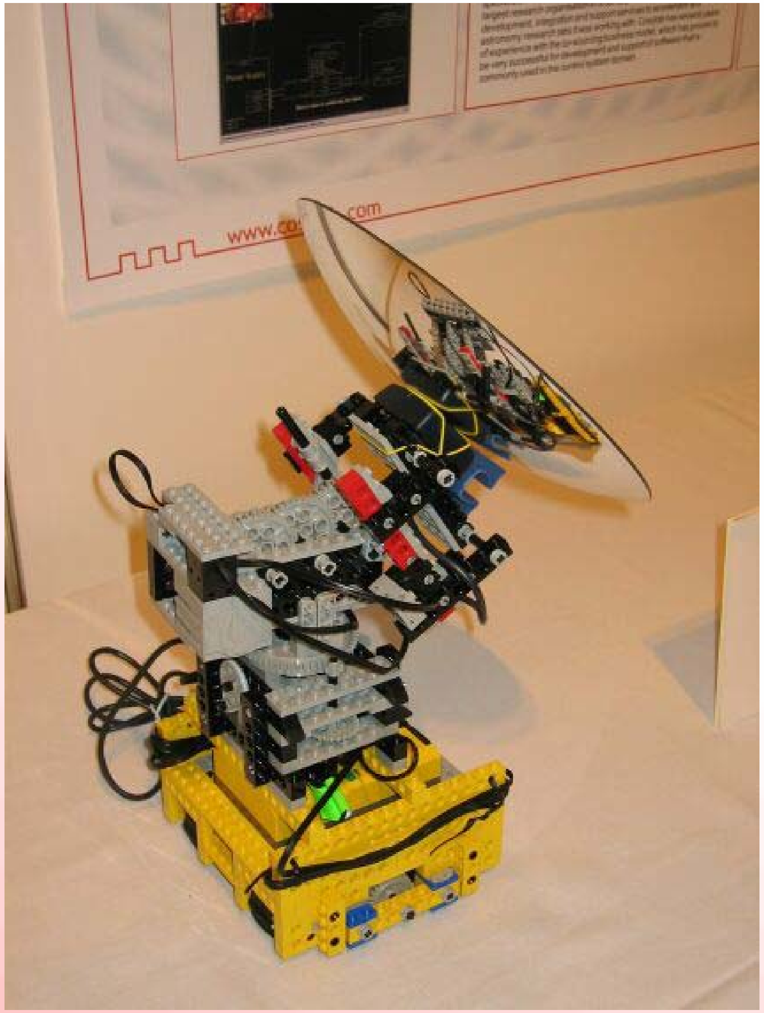
1- Geographic Data Support Ltd (UK)

2- Infoterra Ltd (UK)

18- Hidria (Slovenia)

19- IskraTel Telecomunication (Slovenia)

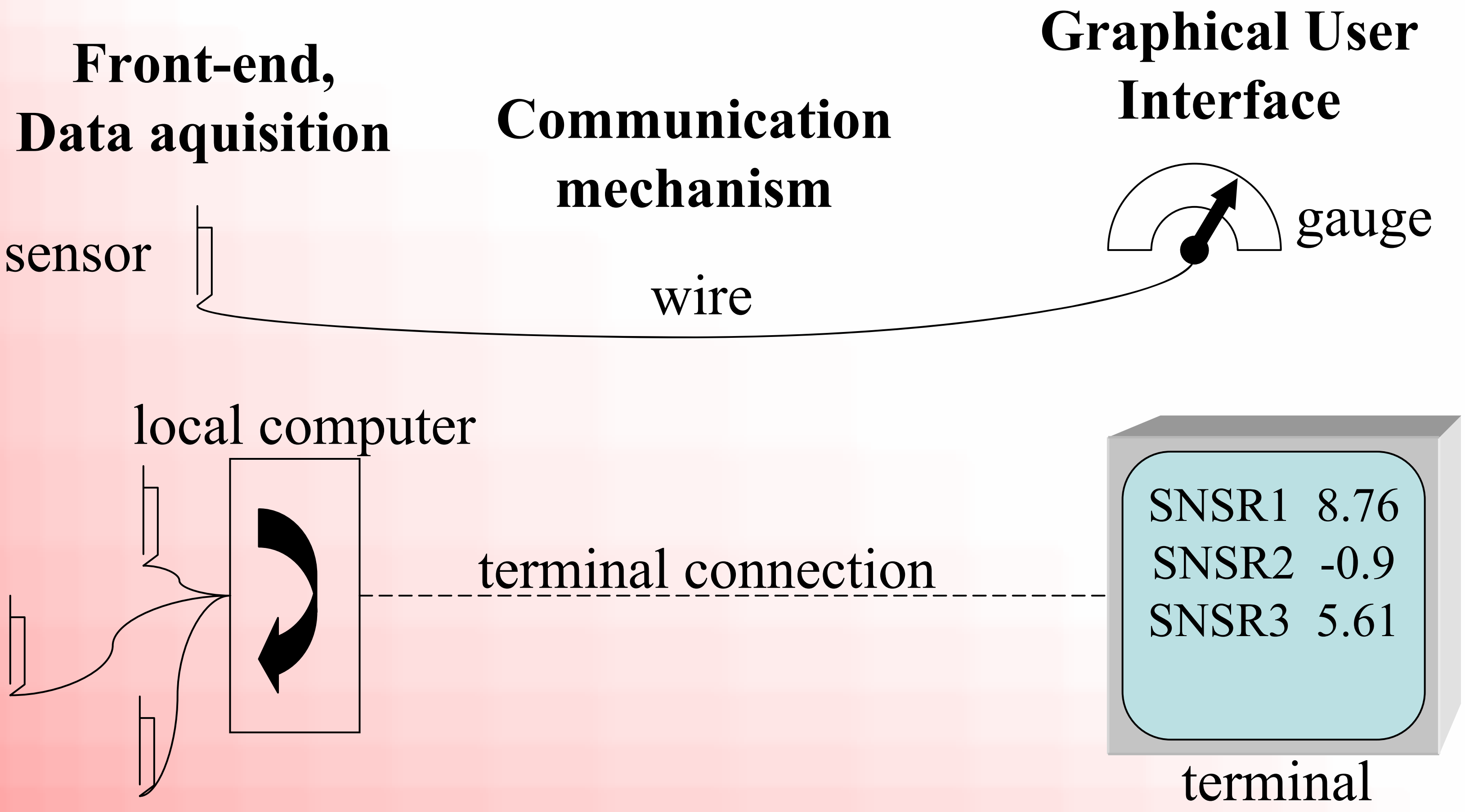
We can sell also other control systems ...



What Is Meant By “Control System”

- Not a shrink-wrap package with an installation wizard, but rather a **service**
  - Engineering according to specifications
  - Configuration of packages like EPICS, TANGO or ACS
    - Some hope this is just a few days of work
  - Outsourcing software/hardware development
  - Installation
    - Some believe this refers to cabling
- All customized for a specific accelerator

From the Analog to the Digital...



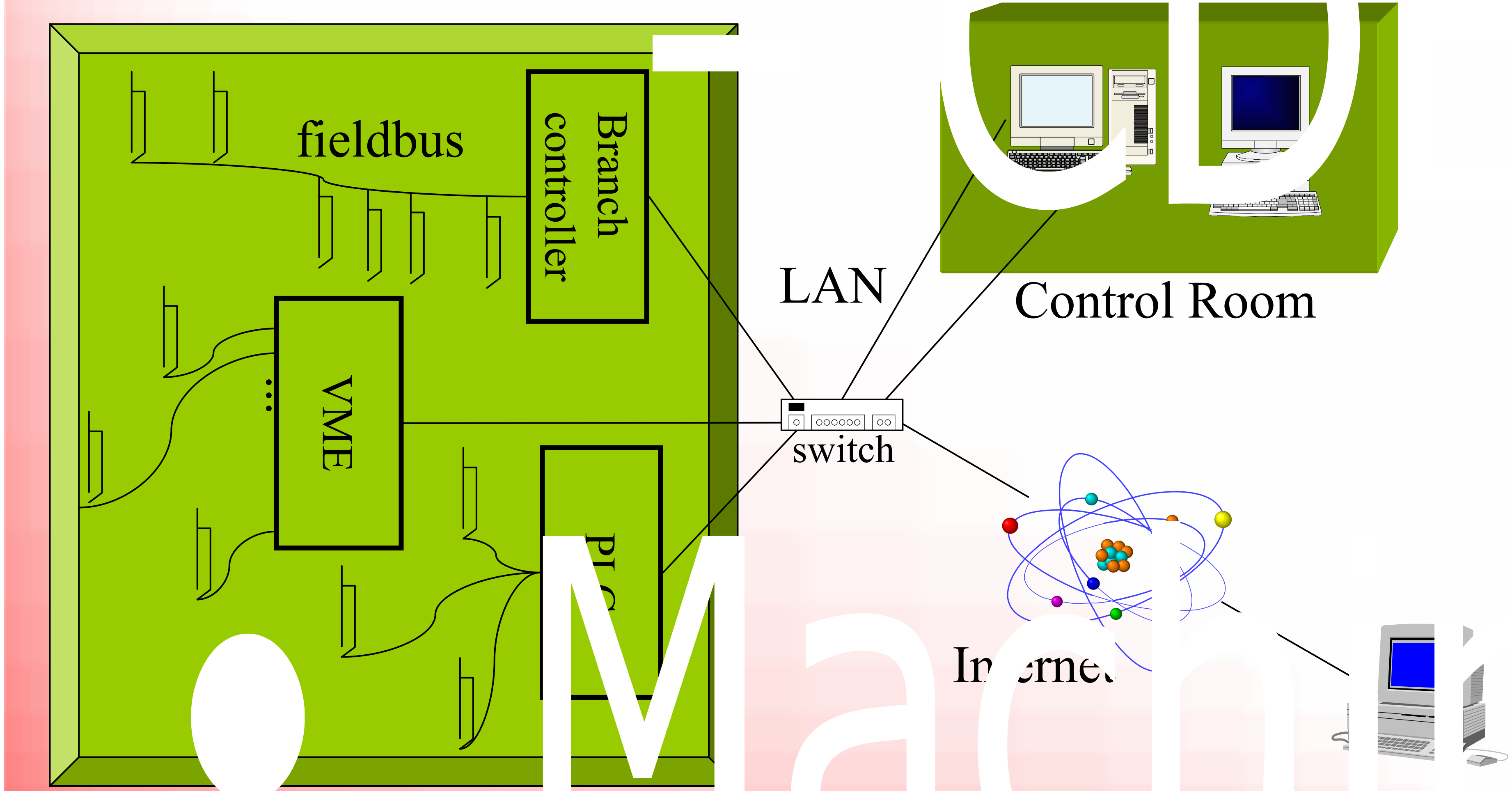
Front-end, Data aquisition

Communication mechanism

Graphical User Interface

SNSR1 8.76
SNSR2 -0.9
SNSR3 5.61

... to the Distributed Era



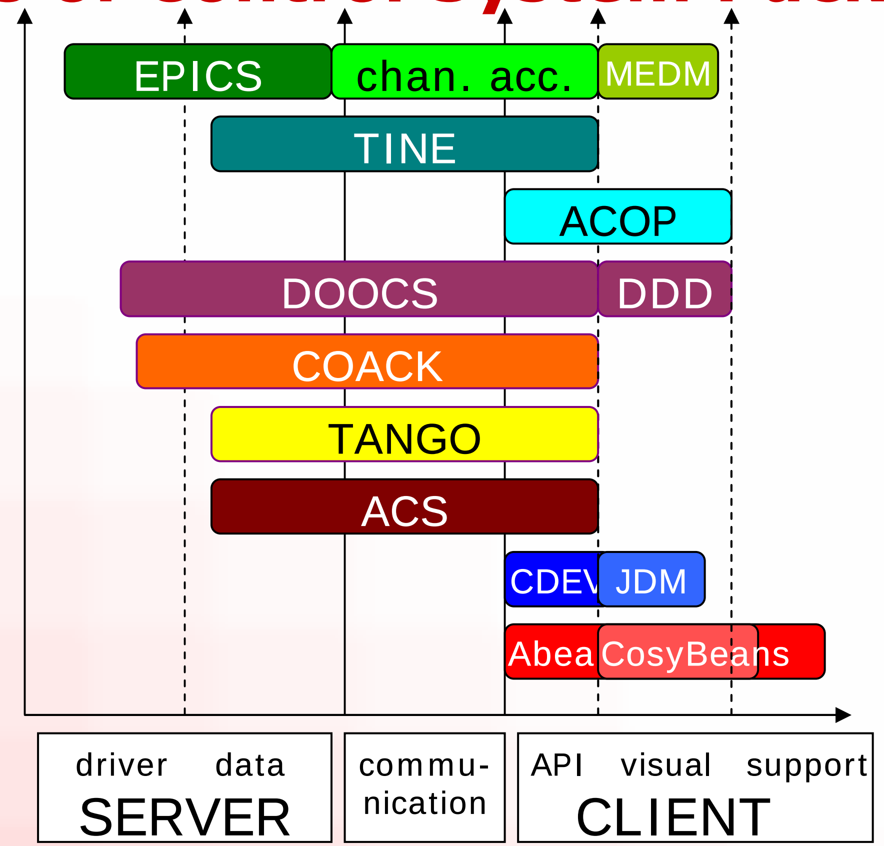
Confused?

- Check this Object Oriented Programming language:
  1. “persistent store”
  2. “method”
  3. “field”
- Translation:
  1. save to file
  2. function, subroutine
  3. variable
- Just don’t trust buzzwords!

Available Control Systems

- Complete systems
  - ACS
  - COACK
  - DOOCS
  - EPICS
  - TANGO
  - TINE
  - ...
- APIs and GUIs
  - Abeans and CosyBeans
  - ACOP
  - CDEV
- Machine physics packages
  - Databush
  - XAL
  - Matlab-based

The Layers of Control System Packages



Applications (alarm manager, GUI, logger, trending, scripting etc.): while all have them, their quality, flexibility, configurable etc. makes the largest difference

Which to Choose?

- Don’t worry:
  - modern computer technology allows any reasonable implementation of software and hardware to function properly
- So what is really important?
  - To define the development procedures
  - To make everyone agree on the interfaces (API)
  - To get the signal list ASAP
  - To get test plan and documentation **before** implementation starts

Development Procedures: Control System is NOT just Playing with Software

- Control Systems are an **engineering** discipline like all the others, but with an even more complicated cycle
  - Write specifications
  - Architecture
  - Design
  - Prototyping – **probably the only fun part**
  - Test procedures
  - Implementation (coding) – **the only software part**
  - Documentation
  - Testing
  - Debugging
  - Acceptance at customer

What a Project Leader Should Ask From Control Group

- Signal List
  - Signal names (define name convention and stick to it – allow no exceptions)
  - Alarm levels and operational limits
- Configuration management
- Logistics of installations
- Error handling
  - How the system behaves when I/O or other errors occur
- Bugs:
  - Plan testing, debugging and workarounds

Why Getting the Control System from Industry?

- Would you build the vacuum chamber or the magnets in-house?
- Why not?
  - Too complicated (technically, procedures, volume)
  - Boring (not fun playing)
- Also electronics was built in-house 20-30 years ago but now seldomly
- What's so different about the control system then?
  - It can be changed arbitrary number of times?
  - It can't be described by a Hamiltonian!

In-house or Outsourcing?

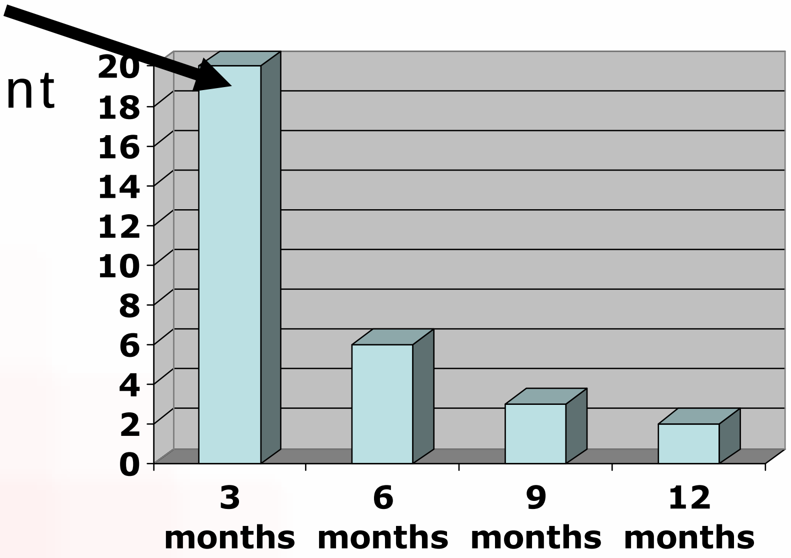
- For in-house: maintenance, upgrades
- Wrong!
- In-house people are smart: but get N different solutions
- Nobody is writing documentation unless forced
  - “Outsourcer” is forced, because of payment
  - In-house person will just tell you, until she/he is gone
- In-house knowhow rests with people, not the lab
- Outsourced knowhow from competent suppliers is like an escrow vault:
  - You pay, but it is well kept for you
  - Over the whole lifetime of the project

The Keyword is Competent Supplier

- What happens if supplier goes bankrupt?
  - A good and honest supplier doesn’t do this
  - Escrow: get all sources at delivery
  - Buy out his people – they know you best!
- Offers standard solutions, well tested **optimized procedures** and **project management**
  - Local scientist-developers have all excellent solutions, but all slightly unique and different
  - We usually deliver more than internal people, just because we know that we get only paid at the end!
- Understands accelerators
  - Programming or automation knowledge is not enough

Common Prejudices (I had them, too)

- A company is more expensive
  - Time is money – expensive is what you can't get done!
  - Big effective cost of new people
- In-house people are more efficient
  - No cure – no pay !
- We can do it faster in-house
  - With or without bugs?
  - Beware of 80/20 rule



- A company can let us down
  - We can't afford this in the small community – we'd be dead
- A company just wants money
  - Are you in science to get money? Don't pressume others are.

The Three Phases of Non-outsourcing

1. We will outsource, but we don’t know yet what
2. We have some specs, but we can handle them ourselves
3. We should have outsourced to you, but now we have already invested so much of our work that we can not justify throwing it all away

- Reminds me of unsuccessful dating ☺

Real Problems

- It’s faster to do it than to write specs
  - True, but if you don’t write specs for yourself, you’ll be in trouble later
- Specs, targets are not clear, can’t control cost
  - True, but then also your own cost wouldn’t be under control
  - Let’s make a fixed price contract, if the effort deviates more than by 10-20%, we renegotiate the contract.
- In-house people can fix problems overnight
  - True: keep one person permanently at lab to collect requests and make quick fixes

The Right Way to Outsourcing: Rightsourcing (you name it!)

- start with smaller projects (2-4 man-weeks)
- regular visits or work on-site
- Get benefits from both “in-sourcing” and “out-sourcing”:
  - 1 person on-site (gather requirements, communicate with customers, organize, support, service…)
  - expert team at home, professionally organized and managed
  - Benefits for the lab:
    - pay only <u>one</u> person, get an expert in <u>every</u> area
    - scientists retain the established work practice: (almost) no specs, creative academic environment, ask and get (almost) next day
    - value for money (efficiently managed, optimized procedures, no cure no pay!)
    - Lifetime support (see what happened at CERN PS)

The Open Source Business Model

- Visual DCT – an important tool for EPICS
  - Used by >50% EPICS users
- No license fees, GPL open source, Cosylab keeps IPR
- Labs paid for improvements
  - Community representative (currently Nick Rees, DLS) collects requirements and feature wish list
  - Nearly 10 contracts of few 10kEUR each
    - Paing lab selects features to be implementes
  - Support contracts under negotiation
    - For ongoing bug fixing

Conclusions

- Why did the spin-off succeed?
  - The right team with the right attitude
- Can the control system be bought from industry?
  - yes, but...
  - you must first choose the right company, one with good understanding of accelerators and with proven competence
- What's in the Future:
  - Software and Control will go where Electronics went

"Do what you do best
and leave to us the rest"