Strategy in architecting

Ronald Fabel
Océ R&D

October 4, 2005

Statement

- **Dana Bredemeyer** (www.bredemeyer.com)
  - Architecture: Translation of strategy into technology
  - Architect: Minimalist

- **<u>One</u> well formulated *strategy* is worth more than 1000 quantifiable *requirements.***

Prove?

- **Océ products**
- **Description of the project and product**
- **Software bottleneck**
- **Software reference architecture in 5 steps**
  - Discussion

Océ products

Office

Color press

- Archiving
- Workflow management
- Document management

Wide format

Transaction printing

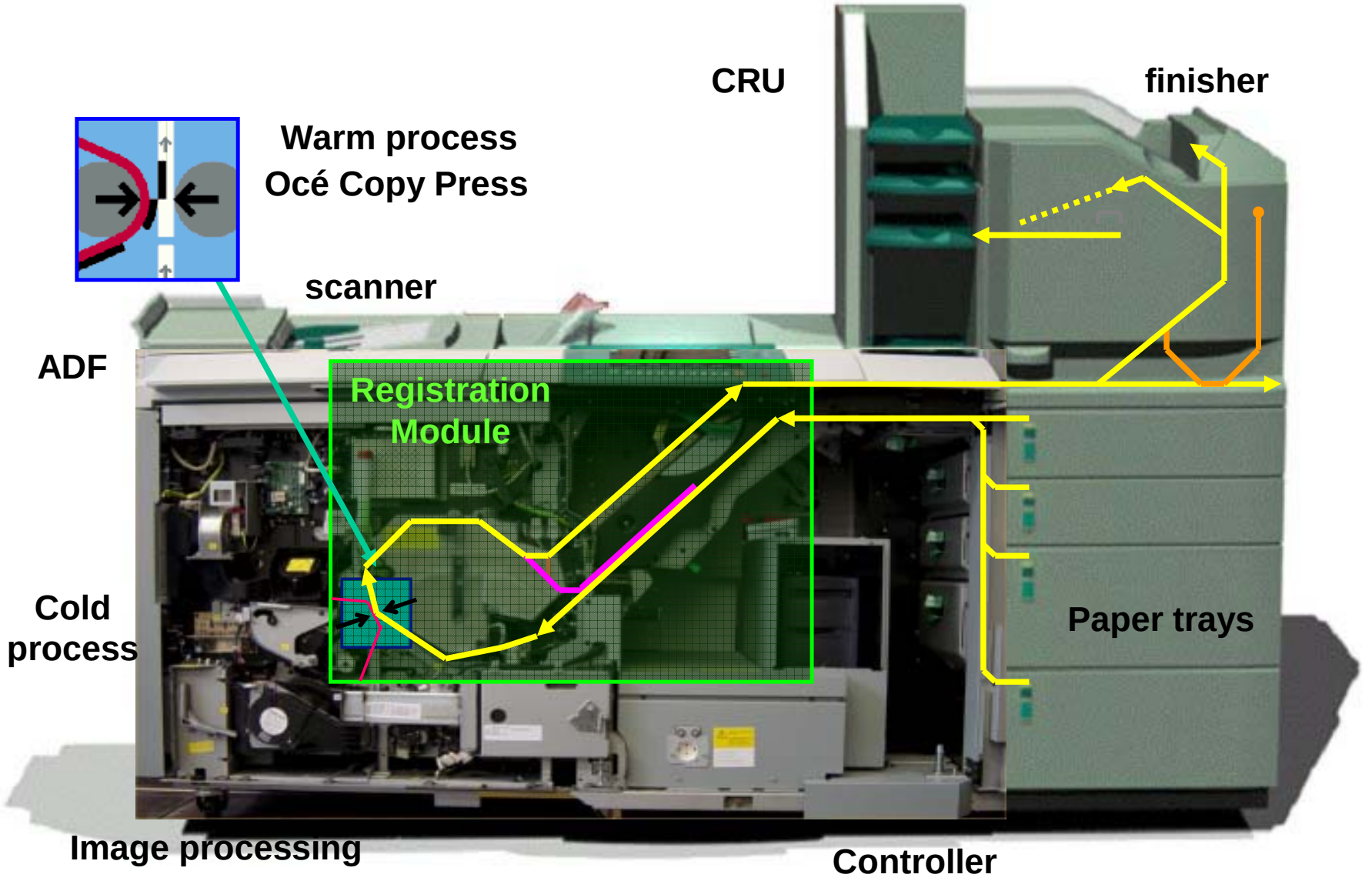
VarioPrint 2090

Project organization

- **Function groups**
  - Paper trays
  - Registration Module
  - Finishing
  - Receiving
  - Cold process
  - Warm process
  - Scanning
  - Document feed
  - Image processing
- **Total construction**
  - Informatics (sw architecture, DME, behavior)
  - Electronics (power, energy, technology)
  - Mechanics (timing, construction)
  - Procede (print/copy quality)

- **Risk areas**
  - I.e: acquisition of print-head
  -

VarioPrint 2090



Bottleneck (1998)

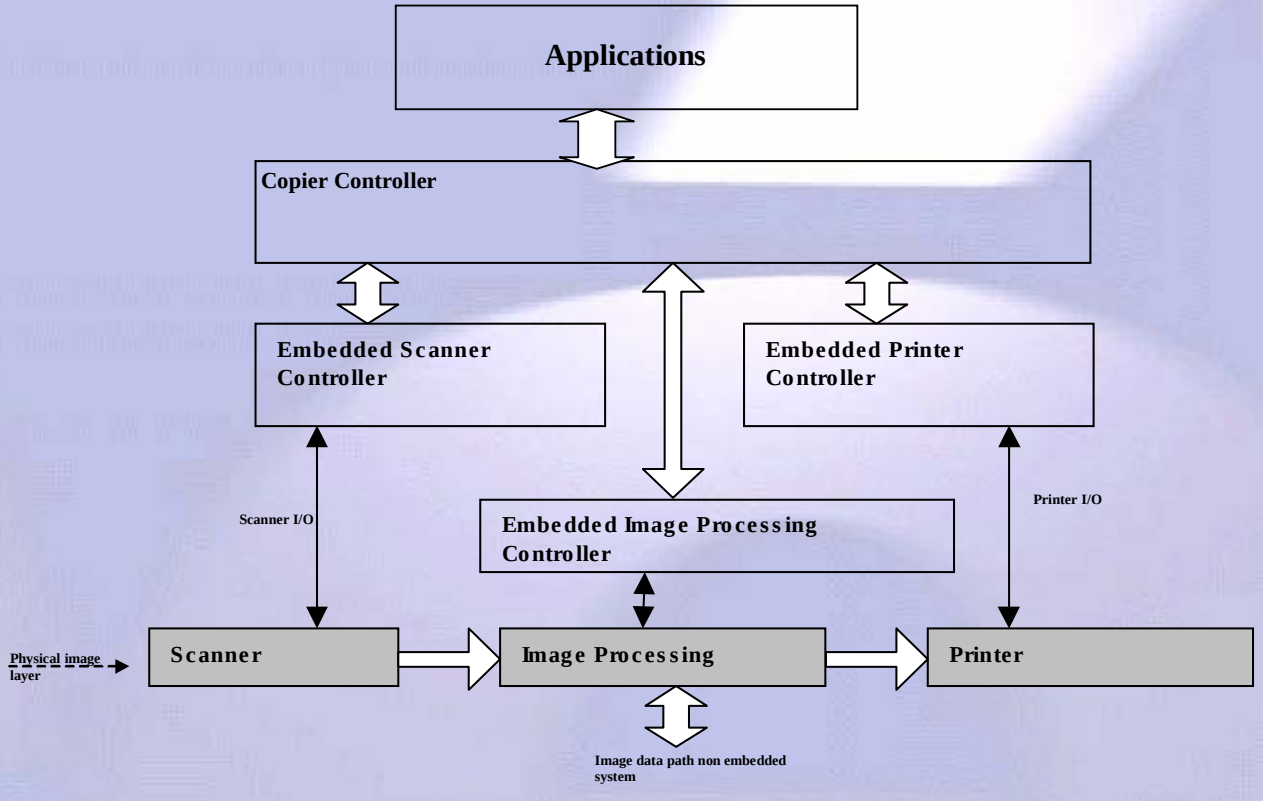
- **Experience hard to get**
- **Introduction of ROOM**
- **Prototype software in C**
- **Change of product profile**
- **SPI and more formal procedures**
- **Attention for requirement management**
- **Rise of software architecting**

- *Many detailed requirement documents*
- *Modularity as a goal*
- *Endless break-down of software components*
- *Layering and abstraction*
- *Every engineer is an architect*
- *Performance problems*
- *Only software engineers occupied with problem solving*

Basis for Software Reference Architecture

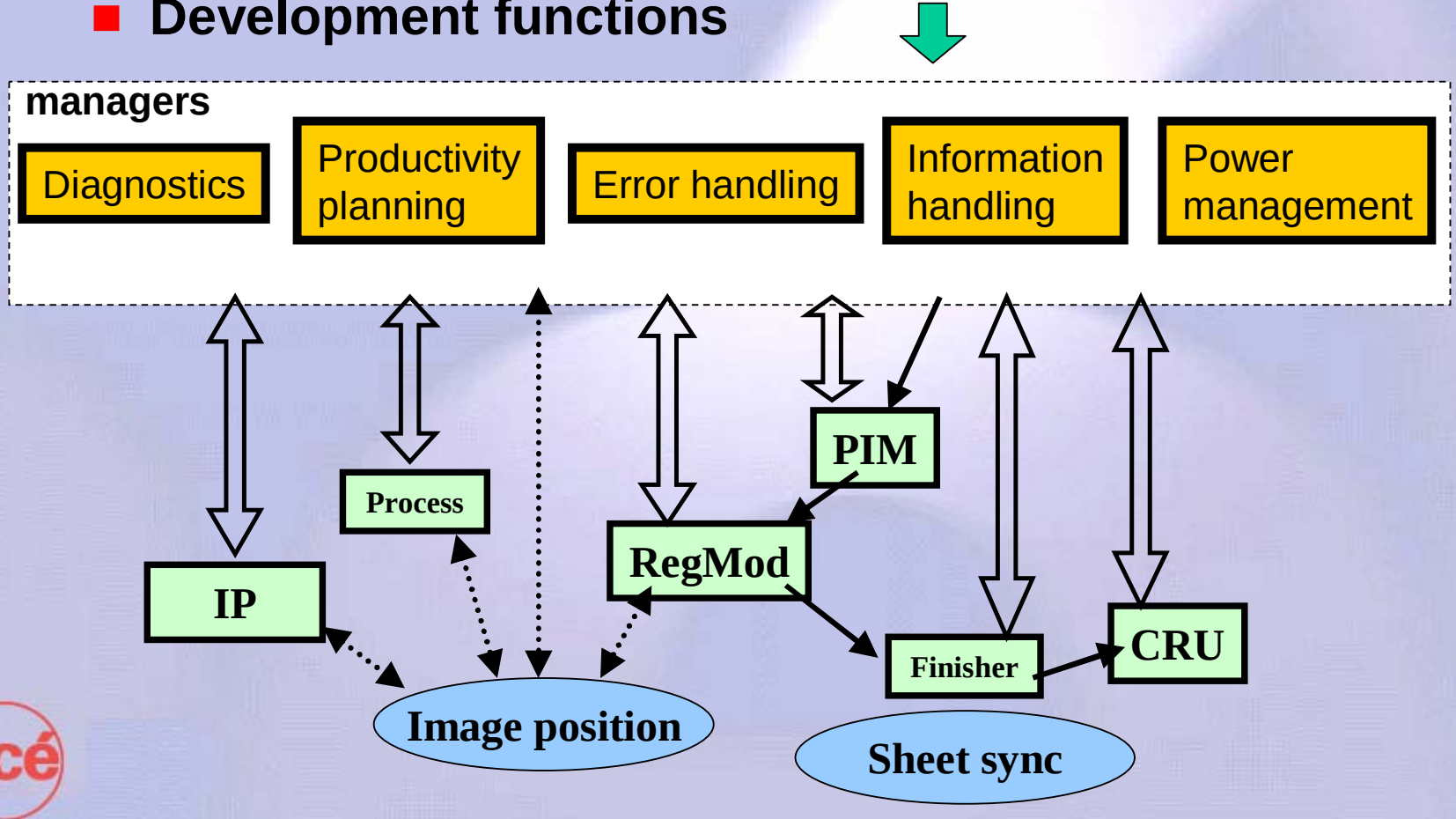
Software architecture 1

- **Page scanners and page printers modules**



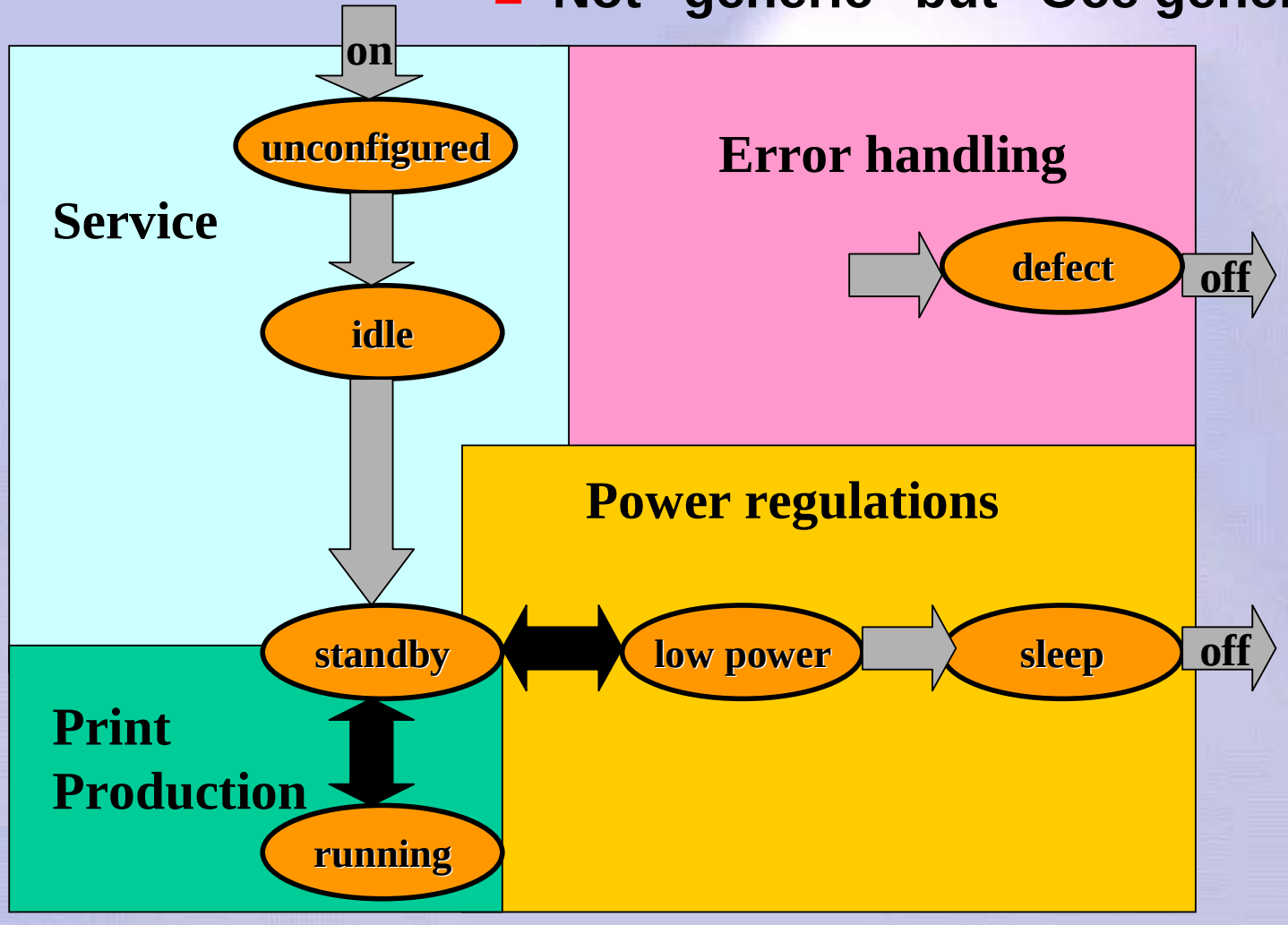
Software architecture 2 (page printer)

- **Usage circumstance**
- **Development functions**



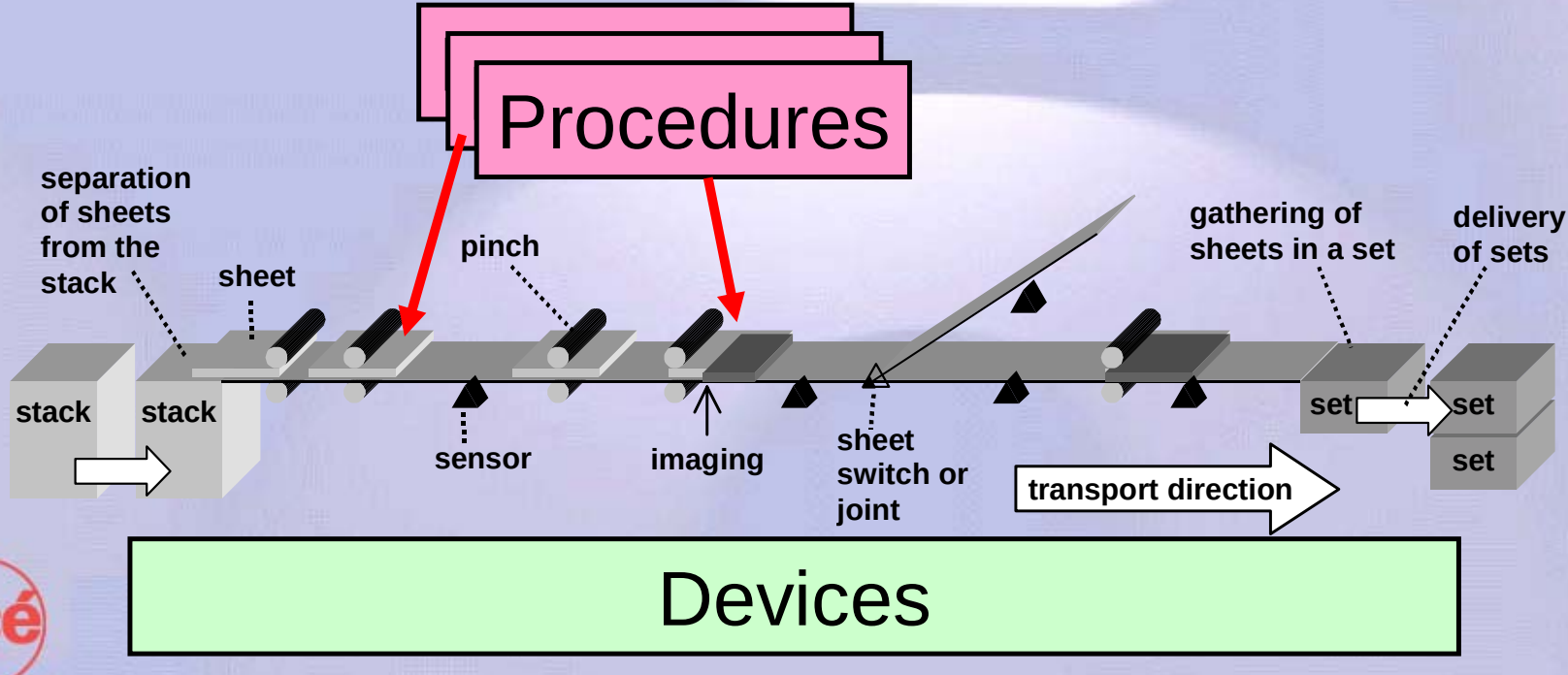
Example status manager

- Not “generic” but “Océ generic”



Software architecture 3

- **Support of multi-disciplinary communication**
  - Behavioural procedures such as transport of a sheet
  - Mechatronic devices such as Z registration unit

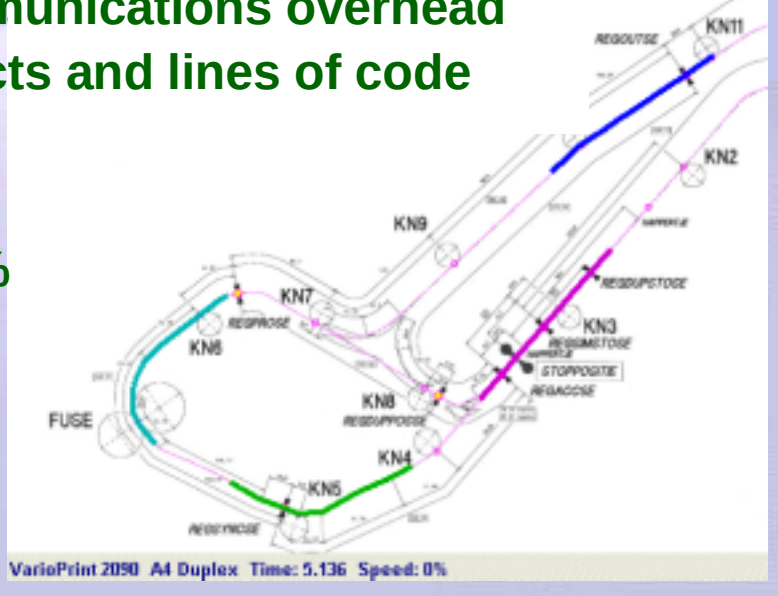


Software architecture 4

- **“Lean and mean”**
  - modularity with a purpose
  - layering with a purpose
  - dynamic configurability with a purpose

  - **Reduction of number of objects**
  - **Reduction of communications overhead**
  - **Reduction of objects and lines of code**

  - **A reduction of 80% in the registration module**



Software architecture 5

- **Timing Backbone**
  - Time triggered architecture for hard real-time functions
  - Distributed deployment along the line of functions to support scalability in a development phase and integration in a engineering phase
  - Performance is under control at each stage of the project

Reference architecture results (2005)

- **Focus on multi-disciplinary innovation**
- **No bottleneck in development nor engineering**
- **Successful reuse of software components**
- **SW community ‘owns’ the timing and behaviour**
- **Keep up with fast hardware roadmaps**

Statement

- **<u>One</u> well formulated *strategy* is worth more than 1000 quantifiable *requirements.***

- ***Strategy titles:***
  - ***“ Make the usage circumstances visible ”***
  - ***“ Embedded means ‘lean and mean’ “***
  - ***“ Time and performance backbone “***
  - ***“ Mechanical engineer should understand the top level software design “***
  - ***“ Modularity needs to serve a purpose “***