Océ Development Processes

Ed Brinksma, ESI

presentation sources:
• Ron Notermans (VP R&D Océ)
• Peter van den Bosch (developer Océ)
• Lou Dohmen (architect Océ)
• Roelof Hamberg (formerly Océ)
• Gerrit Muller (BoDERC project)
Océ - Document Systems

Océ enables its customers to manage their documents efficiently and effectively by offering innovative print and document management products and services for professional environments.

- Consultancy
- Facility Management
- Maintenance

Application Software + Machines

Access

Services software
Principle of working

Network

original

scanning

Network Controller

Image Processing

Control

Transfer & Fuse

Toner

OPC

Paperhandling

Paper

Print

Copy

Paper
Mechanics, Electronics, Procédé
Embedded Software

control
network controller
control
control
Current multidisciplinary approach

- Multidisciplinary project teams (M, E, I, ID, Φ, C, …)
- Common project goal
- Dedication (1 project per person at a time)
- Co-location (1 location per project team)
- Project organisation processes
  - Groups per function, following constructional decomposition
  - TC decides (function leaders, architect(s), project leader)
- Lot of prototyping & testing
  - FunctionTest → Labmodel → EngineeringProtoType
- Years of experience

, but...
Mechanics

• Agenda
  – Cost price
  – Reliability
  – Performance
  – Precision

• Models
  – 3D CAD
  – Total construction drawings
  – Some local Matlab models
  – Specification of time/position tables in Excel

Electronics

• Agenda
  – Cost price
  – Energy consumption
  – Life-cycle-management of components
  – Safety and EMC regulations

• Models
  – System architecture diagrams
  – Circuit simulations (VHDL, PSPICE, Matlab)

Embedded Software

• Agenda
  – Specifications for normal and exceptional behaviour
  – Maximum performance for all jobs
  – Architecture
    • Modular
    • Maintainable
    • Re-use of components

• Models
  – Use Cases
  – UML
  – Text documents
De facto decision process

Product requirements

Mechanics & Physical choices
- Procédé,
- Machine layout,
- Modules,
- Motors,
- Speeds,
- Precision,
- Location of sensors

Electronics choices
- CPU, Memory,
- Databus,
- I/O and Sensors,
- Power,
- Cooling

Software choices

Prototypes
Problem characteristics

- **Development of complex products with**
  - highly interactive sub-domains,
  - where only few people master both ends
  - of the physical and the software world

- **Long lead times**
  - Sequential order of work
  - Coincidental “just-in-time” delivery by hard work
  - Need for more predictability and shorter lead times

- **Software is often on the critical path**
  - Impact of software in earlier phases
  - Integral system approach is required
The Boderc project (2002 – 2006)

High level design method

1. Preparation of the design
   a) Identify (customer) key drivers and requirements
   b) Identify realization aspects of concern
   c) Consolidate core domain knowledge

2. Selection of critical design aspects
   a) Identify tensions and conflicts (qualitative)
   b) Gather facts and identify uncertainties to quantify tensions and conflicts

3. Evaluation of design aspects
   a) Build small models
      (small = hours to 4 weeks of effort)
   b) Perform measurements
Example step 2a)
Identify tensions & conflicts (qualitative)
Example step 3a)
Build small models: Printer topology & sheet flow
Example step 3a)
Build small models: Semi-Static Power Decomposition

± 2000 W

Mains power available for the print-engine

± 15%

Print-engine control:
* User interface:
  - Display
* Datapath:
  - Printer controller
  - Image data processing
* Embedded controller:
  - Main and local nodes

± 10%

Scanner:
* Document feeder
* Scan electro

± 10%

Motors Paper Path:
* Paper input section
* Registration section
* Procédé section
  - Transfer system
  - Imaging system
* Turn section
* Paper output section

± 65%

Transfer unit

Preheating unit

Low-Voltage Supply
η = 85%

High-Voltage Supply
η = 95%

Cooling System:
* Imaging Unit cooling
* Cleaner cooling

Imaging unit

Cleaner

Paper

Total power dissipation
Example step 3b)  
Measure: Dynamic behavior motors paper path

![Graphs showing voltage and power measurements over time](image-url)
Iteration: dynamic workflow model

Preparation
- identify key drivers & requirements
- critical realization aspects

Drivers
- data, models, experience

Changes
- key requirements realization options
- questions

Selection of critical aspects
- identify tensions and conflicts
- gather facts & uncertainties

Evaluation of design options
- build small models
- perform measurements

Design
- Record Design decisions

November 12th, 2008 Océ Development Process
Challenges:

- Current systems are developed to operate reliably within a **very well specified range** of conditions (e.g. temperature, energy, humidity, etc)
- Such systems often do **not easily adapt** to environmental or product usage fluctuations

- An approach is to design and implement rapidly; equivalent to “next release”
  - Essential to **understand** the effects of design changes
- Another approach is **system’s adaptability**
  - Adapt the actual use of the product at run-time
Octopus objectives

- At **design-time**, how
  - to build systems that are kept in a predictable, reliable, and operational mode,
  - while performing their task in an adapted manner?

- At **run-time**, how
  - to detect in what state the system is,
  - in what direction to adapt the product behavior in its operational space, and,
  - how to put this adaptability into effect?
1. Virtual prototypes

1. Make good predictions early in the design phase
2. Develop integral models that cover medium and paper path, mechanics, black-box view of print process, etc.
   - including performance, resource usage, energy, space, …
3. Experiment with model without expensive (e.g. time-consuming) prototypes
4. Gradually replace models with real components
5. Tightly link architectural and design trade-offs
6. Validate integral models
Research approach Octopus (2)

2. Adaptivity

Input:
- Document
- Media

Output:
- Document
- User satisfaction

Design-time adaptivity

Run-time adaptivity

System observer:
1. What happens, when, how
2. Detect problems
3. System diagnosis
4. Possible solution

R&D

On-site service adaptation

Fleet Database
Research approach Octopus (3)
several practical cases

- **Bayesian networks**
- **Pareto optimality**
- **adaptive control**
- **learning**

**Demonstrator PQ 1**
Trade off power consumption and productivity

**Demonstrator PQ 2**
Multiple trade-offs including heat and glossiness

**Demonstrator PQ 3**
Full print quality control incorporating physical control and data-path

**Case PQ1**
Print quality

**Case PQ2**
Data path

**Case PQ3**
Software architectures

**Case DP1**
High performance multiple scenario data path without adaptability

**Case DP2**
Data path without adaptability

**Demonstrator DP1**
PetriNets

**Demonstrator DP2**
Uppaal

**SDF**