Module 20 Medical Imaging case, CAFCR illustration

by Gerrit Muller    Buskerud University College

    e-mail: gaudisite@gmail.com

www.gaudisite.nl

Abstract
This module provides a complete illustration of the CAFCR based architecting method. The case is a Medical Imaging Workstation, created in the early nineties.
Abstract
The chronological events of the product creation of the medical imaging workstation are discussed. The growth in functionality and size from prototype to product is shown. Typical problems in this period are explained.
Chronology of Easyvision RF R1 development

1991
- basic application
- toolboxes
- 100 kloc
- interactive viewing

1992
- performance problems
- IQ problems

1993
- Easyvision RF integrated product
- 360 kloc
- print server + communication + interactive viewing

marketing opinion:
"All the functionality is available, we only have to provide a clinical UI"
Medical Imaging R/F

Print Store View Cluster

Spool HCU Store Image Gfx UI DB PMS-net in PMS-net out

RC driver HC driver DOR driver

NIX

SunOS, SunView

Standard Sun workstation

Desk, cabinets, cables, etc.

Medical Imaging in Chronological Order

5 Gerrit Muller

version: 1.2
July 31, 2014
MITORswLayers1991
Solution of memory performance problem

-measured
code
OS
data
bulk data
fragmentation
anti-fragmenting
budget based
awareness,
measurement
DLLs
tuning

budget

200 MB

74 MB
Visualization memory use per process

- UNIX
- shared libraries
- communication server
- storage server
- print server
- other

Measured data (left column) vs. budget per process (right column) in MByte:
- Code: 20
- Code (uncertainty): 10
- Data: 30

Graph: Bar chart showing resource usage across different categories.
Causes of performance problems other than memory use

- remote systems and users
- communication
- user interface
- user
- graphics updates
- framebuffer access
- UI devices
- data base granularity
- information model layering
- process communication overhead
- active data granularity, update
- network I/O overhead
- file I/O overhead
- export
- network I/O overhead
- disk drive
- optical storage
- optical disk drive
- print
- printer
- I/O overhead

Medical Imaging in Chronological Order

version: 1.2
July 31, 2014
MITORperformanceCauses
Image quality and safety problem

false contour

\( f(x) \)

10 bits pixel value
8 bits pixel value

\( x \)
for user readability the font-size was determined "intelligently"; causing a dangerous mismatch between text and image

- URF monitor output: fixed size letters at fixed grid
- EV output: scaleable fonts in graphics overlay
Abstract
The Customer objectives, Application and Functional views are described. The radiology department and the radiologist are the main customer. The clinical and the financial context of the radiology department is shown. The medical imaging workstation is positioned in the field of IT products and in the clinical workflow.

The market segmentation is shown. The typical URF examination is explained. Key drivers are linked to application drivers and to product requirements. The functionality development over time is shown and the role of the information model for interoperability is discussed.
The clinical context of the radiology department

- Family doctor
- Referring physician
- Radiologist
- Nurse, operator
- Patient
- Request
- Report
- Findings
- Image
- Film

Legend:
- Paper or el. form
- Electronic
- Human interaction
- Intense
- Weak

Medical Imaging Workstation: CAF Views
version: 1.4
July 31, 2014
MICAFRequestFlow
The financial context of the radiology department

- Insurance
- Government
- Patient
- Radiology
- Hospital Administration
- Equipment and Service Providers

Relationships:
- Payment
- Budget
- Bill
- Regulations
- Facilities
- Schedules
- Equipment
- Services
- Payment
Application layering of IT systems

- modalities from other vendors
- Philips modalities
- medical imaging workstation
- archive, report, review, tele, print
- PACS
- HIS, RIS, CIS, LIS
- IT infrastructure
- administrative mainframes
- legend
  - modality systems
  - modality enhancement
  - image workflow
  - clinical specific info
  - generic
### Reference model for healthcare automation

#### Information handling
- Entirely distributed.
- Wide variation due to "socio-geographics":
  - Psycho-social,
  - Political, cultural factors.

#### Imaging and treatment
- Localised.
- Patient focus.
- Safety critical.
- Limited variation due to "nature":
  - Human anatomy,
  - Pathologies,
  - Imaging physics.

#### Image handling
- Distributed.
- Limited variation due to "nature":
  - Human anatomy,
  - Pathologies,
  - Imaging physics.

#### Base technology
- Not health care specific.
- Short life-cycles.
- Rapid innovation.

#### Archiving
- Service business.
- Not health care specific.
- Extreme robustness.
- Fire, earthquake, flood proof.
- Life time: 100 yrs (human life).

---

Medical Imaging Workstation: CAF Views

Gerrit Muller

Version: 1.4
July 31, 2014
MICAFReferenceModel
Clinical information flow

- Acquire images
- Prepare diagnosis
- Diagnosis
- Report authorise
- Archive

Clinical value
Richness

Time

Clinical review
Education
Research
Treatment planning
Demonstration

Medical imaging workstation

Medical Imaging Workstation: CAF Views
17 Gerrit Muller
Typical case URF examination

3 examination rooms connected to

1 medical imaging
workstation + printer

examination room: average 4 interleaved examinations / hour

image production: 20 1024 ² 8 bit images per examination

film production: 3 films of 4k*5k pixels each

high quality output
(bi-cubic interpolation)
Timing of typical URF examination rooms

9:00 - 10:00

Patient 1: Exam Room 1
Patient 2: Exam Room 2
Patient 3: Exam Room 1
Patient 4: Exam Room 2

1 hour
### Key drivers, application drivers and requirements

<table>
<thead>
<tr>
<th>Customer key drivers</th>
<th>derived Application drivers</th>
</tr>
</thead>
<tbody>
<tr>
<td>report quality</td>
<td>selection of relevant material use of standards</td>
</tr>
<tr>
<td>diagnostic quality</td>
<td>acquisition and viewing settings contrast, brightness and resolution of light-box</td>
</tr>
<tr>
<td>safety and liability</td>
<td>clear patient identification left right indicators follow procedures freeze diagnostic information</td>
</tr>
<tr>
<td>cost per diagnose</td>
<td>interoperability over systems and vendors multiple images per film minimise operator handling multiple applications per system</td>
</tr>
<tr>
<td>time per diagnose</td>
<td>diagnose at light-box with films all preparation in exam room</td>
</tr>
</tbody>
</table>

#### Requirements

- import
- auto-print
- parameterized layout
- spooling
- storage
- navigation / selection
- auto-delete
- viewing
- contrast / brightness
- zoom
- annotate
- export

- functionality
- system response
- system throughput
- image quality
- annotation
- material cost
- operational cost
- qualities

- shared information model
- viewing settings
- patient, exam info

- interfaces
Information model, standardization for interoperability

High innovation rate

Global standardization takes more than 5 years

High interoperability

legend
- applications
- product family
- vendor
- world standard

ACR/NEMA | DICOM
---|---
Siemens | GE | Philips

CT | MRI

cardio vascular | URF | medical imaging

bolus chase | vascular analyse | RF

cardio analyse
Coverage of submethods of the CAF views

<table>
<thead>
<tr>
<th>Customer objectives</th>
<th>Application</th>
<th>Functional</th>
</tr>
</thead>
<tbody>
<tr>
<td>key drivers</td>
<td>context diagram</td>
<td>case descriptions</td>
</tr>
<tr>
<td>value chain</td>
<td>stakeholders and concerns</td>
<td>commercial decomposition</td>
</tr>
<tr>
<td>business models</td>
<td>entity relationship models</td>
<td>service decomposition</td>
</tr>
<tr>
<td>suppliers</td>
<td>dynamic models</td>
<td>goods flow decomposition</td>
</tr>
<tr>
<td></td>
<td></td>
<td>function and feature</td>
</tr>
<tr>
<td></td>
<td></td>
<td>specifications</td>
</tr>
<tr>
<td></td>
<td></td>
<td>performance</td>
</tr>
<tr>
<td></td>
<td></td>
<td>external interfaces</td>
</tr>
<tr>
<td></td>
<td></td>
<td>standards</td>
</tr>
</tbody>
</table>

Legend:
- *explicitly addressed*
- *addressed only implicitly*
- *not addressed*

Coverage based on documentation status of first product release

Medical Imaging Workstation: CAF Views
Gerrit Muller

version: 1.4
July 31, 2014
MICAFconclusion
Abstract
The concepts and realization of the medical imaging workstation are described. The following concepts are described: presentation and processing pipeline, resource management (CPU and memory), including caching and anti-fragmentation strategy, software process decomposition and decomposition rules.

The actual realization figures serve as illustration for the justification of some of the concepts.
Image Quality expectation WYSIWYG

what you see at one work-spot is what you get at another work-spot

X-ray system

application processing

Easyvision

3rd party workstation

monitor

film

network, storage

monitor

film

network, storage

version: 2.7
July 31, 2014
MICVwysiwyg
Presentation pipeline for X-ray images

1. Image from database
2. Spatial enhancement: bi-linear, bi-cubic
3. Interpolate
4. Look up table (LUT) with invert, contrast, brightness
5. Graphics merge
6. Colour LUT
7. Monitor

Legend:
- SW: Software
- HW: Hardware
Quadruple view-port screen layout

UI icons, text

view-port 1
view-port 2
view-port 3
view-port 4
view-port 5

1152 pixels
960 pixels
ca. 460 pixels
ca. 200 pixels

1152 pixels

960 pixels

c. 200 pixels

c. 460 pixels

Medical Imaging Workstation: CR Views
Gerrit Muller
version: 2.7
July 31, 2014
MICVquadrupleViewportLayout
**Rendered images at different destinations**

**Screen:**
- low resolution
- fast response

**Film:**
- high resolution
- high throughput

**Network:**
- medium resolution
- high throughput
Concurrency via software processes

- remote systems and users
- communication
- user
- user interface
- data base
- export
- network
- disk drive
- optical storage
- optical disk drive
- print
- printer

Legend:
- client
- client process
- server process
- operational process
- control and data flow
- user control
- UI devices
- system monitor
- Unix daemons
- associated hardware

Medical Imaging Workstation: CR Views
30  Gerrit Muller
version: 2.7
July 31, 2014
MICVsoftwareProcess
Criterions for process decomposition

- management of concurrency
- management of shared devices
- unit of memory budget (easy measurement)
- enables distribution over multiple processors
- unit of exception handling: fault containment and watchdog monitor
### Simplified layering of the software

<table>
<thead>
<tr>
<th>Medical Imaging R/F</th>
<th>Print</th>
<th>Store</th>
<th>View</th>
<th>Cluster</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spool</td>
<td>HCU</td>
<td>Store</td>
<td>Gfx</td>
<td>UI</td>
</tr>
<tr>
<td>RC driver</td>
<td>HC driver</td>
<td>DOR driver</td>
<td>NIX</td>
<td>SunOS</td>
</tr>
</tbody>
</table>
| RC interf | HC interf | DOR    |       | Desk, cabinets, cables, etc. |}

#### legend
- **user interface**
- **application functions**
- **toolbox**
- **operating system**
- **hardware**
- **SW infrastructure**
- **connected system**

---

**Medical Imaging Workstation: CR Views**

32  Gerrit Muller

version: 2.7
July 31, 2014
MICVswLayers1992
## Memory budget of Easyvision RF R1 and R2

<table>
<thead>
<tr>
<th>memory budget in Mbytes</th>
<th>R1</th>
<th>R2</th>
<th>R1</th>
<th>R2</th>
<th>R1</th>
<th>R2</th>
<th>R1</th>
<th>R2</th>
</tr>
</thead>
<tbody>
<tr>
<td>shared code</td>
<td>6.0</td>
<td>11.0</td>
<td>2.2</td>
<td>4.2</td>
<td>15.4</td>
<td>0.5</td>
<td>28.5</td>
<td>7.0</td>
</tr>
<tr>
<td>UI process</td>
<td>0.2</td>
<td>0.3</td>
<td>2.0</td>
<td>3.0</td>
<td>12.0</td>
<td>12.0</td>
<td>14.2</td>
<td>15.3</td>
</tr>
<tr>
<td>database server</td>
<td>0.2</td>
<td>0.3</td>
<td>4.2</td>
<td>3.2</td>
<td>3.0</td>
<td>4.4</td>
<td>6.5</td>
<td></td>
</tr>
<tr>
<td>print server</td>
<td>0.4</td>
<td>0.3</td>
<td>2.2</td>
<td>1.2</td>
<td>7.0</td>
<td>9.0</td>
<td>9.6</td>
<td>10.5</td>
</tr>
<tr>
<td>DOR server</td>
<td>0.4</td>
<td>0.3</td>
<td>4.2</td>
<td>2.0</td>
<td>2.0</td>
<td>1.0</td>
<td>6.6</td>
<td>3.3</td>
</tr>
<tr>
<td>communication server</td>
<td>1.2</td>
<td>0.3</td>
<td>15.4</td>
<td>2.0</td>
<td>10.0</td>
<td>4.0</td>
<td>26.6</td>
<td>6.3</td>
</tr>
<tr>
<td>UNIX commands</td>
<td>0.2</td>
<td>0.3</td>
<td>0.5</td>
<td>0.2</td>
<td>0.2</td>
<td>0.7</td>
<td>0.5</td>
<td></td>
</tr>
<tr>
<td>compute server</td>
<td>0.3</td>
<td>0.3</td>
<td>0.5</td>
<td>6.0</td>
<td>6.0</td>
<td>6.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>system monitor</td>
<td>0.3</td>
<td>0.3</td>
<td>0.5</td>
<td></td>
<td></td>
<td>0.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>application total</td>
<td>8.6</td>
<td>13.4</td>
<td>28.5</td>
<td>12.6</td>
<td>31.0</td>
<td>35.0</td>
<td>66.1</td>
<td>61.0</td>
</tr>
<tr>
<td>UNIX file cache</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>7.0</td>
<td>10.0</td>
</tr>
<tr>
<td>total</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>76.1</td>
<td>74.0</td>
</tr>
</tbody>
</table>
Memory fragmentation

1. replace image 3 by image 4

2. add image 5

3. replace image 1 by image 6
Memory fragmentation increase

- used address space
- gross used
- nett used

MBytes vs time

Medical Imaging Workstation: CR Views
version: 2.7
July 31, 2014
Cache layers

Medical imaging R/F cache sizes

- cluster PixMap cache
- print PixMap cache
- view PixMap cache
- allocator, chunk
- heap memory, malloc() free()
- virtual memory
  - memory management unit
  - instruction and data cache
  - physical memory
  - disk storage

Legend

- User interface
- Application functions
- Toolbox
- Operating system
- Hardware
Bulk data memory management memory allocators

- **Chunk size:** 3MB
  - For large images
    - From 225 kB (480*480*8) to 3 MB (1536*1024*16)
- **Block size:** 256kB

- **Chunk size:** 1MB
  - For stamp images
    - 96*96*8 (9kB)
- **Block size:** 9kB

- **Chunk size:** 2MB
  - For small (screen) images
    - From 8kB to 225 kB
- **Block size:** 8 kB

---

Medical Imaging Workstation: CR Views
Gerrit Muller
Cached intermediate processing results
Example of allocator and cache use

Pixmap cache

1024^2 8 bit image requires 4 256kB blocks
8 1024^2 images require 48 256kB blocks 12 blocks shortage

460^2 image 8 bit requires 27 8kB blocks
200^2 images require 5 8kb blocks
all screen-size images require
334 8kB blocks, 78 blocks shortage

viewports
display

version: 2.7
July 31, 2014
MICV pixmap Example
Print server is based on banding

Medical Imaging Workstation: CR Views
40  Gerrit Muller
CPU processing times and viewing responsiveness

- Retrieve: 0.3s
- Enhance: 0.5s
- Interpolate: 0.2s
- Lookup (LUT): 0.075s
- Merge: 0.025s
- Display: 0.05s

Pipeline timing proportional to accumulated processing time in seconds:

- 0.05s
- 0.025s
- 0.075s
- 0.1s
- 0.2s
- 0.3s
- 0.5s
- 0.7s
- 0.8s
- 0.9s
- 1.0s
- 1.1s

Update rate for common user actions:
- Next: 0.9s\(^{-1}\)
- Zoom: 3 s\(^{-1}\)
- C/B: 7 s\(^{-1}\)

Accumulated processing time in seconds:

- 1024\(^2\)
- 920\(^2\)

Medical Imaging Workstation: CR Views

version: 2.7
July 31, 2014
MICVprocessingTimes
Server CPU load

Remote systems and users

Communication

Database

Print

Printer

Disk

Import

Print

3.5 CPU seconds per Mpixel output

2.5 CPU seconds per Mbyte input

50 s/exam

210 s/exam

50 s/exam

210 s/exam

CPU time available for interactive viewing

Margin

2 min

Import

2.5 min / exam

Print

10.5 min / exam

Serving one examination room

Serving 3 examination rooms

Serving 3 examination rooms

Serving one examination room

30%

90%

30%

90%
Resource measurement tools

\[
t_{n-2} \quad \text{preamble to remove start-up effects} \quad t_{n-1} \quad \text{use case} \quad t_n \quad \text{time}
\]

- \text{oit}
- \text{ps}
- \text{vmstat}
- kernel resource stats

\[\Delta \text{ object instantations heap memory usage}\]

- kernel CPU time
- user CPU time
- code memory
- virtual memory
- paging

heapviewer (visualise fragmentation)
## Object Instantiation Tracing

<table>
<thead>
<tr>
<th>class name</th>
<th>current nr of objects</th>
<th>deleted since $t_{n-1}$</th>
<th>created since $t_{n-1}$</th>
<th>heap memory usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>AsynchronousIO</td>
<td>0</td>
<td>-3</td>
<td>+3</td>
<td></td>
</tr>
<tr>
<td>AttributeEntry</td>
<td>237</td>
<td>-1</td>
<td>+5</td>
<td></td>
</tr>
<tr>
<td>BitMap</td>
<td>21</td>
<td>-4</td>
<td>+8</td>
<td></td>
</tr>
<tr>
<td>BoundedFloatingPoint</td>
<td>1034</td>
<td>-3</td>
<td>+22</td>
<td>[819200]</td>
</tr>
<tr>
<td>BoundedInteger</td>
<td>684</td>
<td>-1</td>
<td>+9</td>
<td>[8388608]</td>
</tr>
<tr>
<td>BtreeNode1</td>
<td>200</td>
<td>-3</td>
<td>+3</td>
<td></td>
</tr>
<tr>
<td>BulkData</td>
<td>25</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>ButtonGadget</td>
<td>34</td>
<td>0</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>ButtonStack</td>
<td>12</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>ByteArray</td>
<td>156</td>
<td>-4</td>
<td>+12</td>
<td>[13252]</td>
</tr>
</tbody>
</table>
# Overview of benchmarks and other measurement tools

<table>
<thead>
<tr>
<th>test / benchmark</th>
<th>what, why</th>
<th>accuracy</th>
<th>when</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>public</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SpecInt (by suppliers)</td>
<td>CPU integer</td>
<td>coarse</td>
<td>new hardware</td>
</tr>
<tr>
<td>Byte benchmark</td>
<td>computer platform performance</td>
<td>coarse</td>
<td>new hardware</td>
</tr>
<tr>
<td></td>
<td>OS, shell, file I/O</td>
<td></td>
<td>new OS release</td>
</tr>
<tr>
<td><strong>self made</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>file I/O</td>
<td>file I/O throughput</td>
<td>medium</td>
<td>new hardware</td>
</tr>
<tr>
<td>image processing</td>
<td>CPU, cache, memory as function of image, pixel size</td>
<td>accurate</td>
<td>new hardware</td>
</tr>
<tr>
<td>Objective-C overhead</td>
<td>method call overhead</td>
<td>accurate</td>
<td>initial</td>
</tr>
<tr>
<td></td>
<td>memory overhead</td>
<td></td>
<td></td>
</tr>
<tr>
<td>socket, network</td>
<td>throughput</td>
<td>accurate</td>
<td>ad hoc</td>
</tr>
<tr>
<td></td>
<td>CPU overhead</td>
<td></td>
<td></td>
</tr>
<tr>
<td>data base</td>
<td>transaction overhead</td>
<td>accurate</td>
<td>ad hoc</td>
</tr>
<tr>
<td></td>
<td>query behaviour</td>
<td></td>
<td></td>
</tr>
<tr>
<td>load test</td>
<td>throughput, CPU, memory</td>
<td>accurate</td>
<td>regression</td>
</tr>
</tbody>
</table>
Coverage of submethods of the CR views

<table>
<thead>
<tr>
<th>Conceptual</th>
<th>Realization</th>
</tr>
</thead>
<tbody>
<tr>
<td>construction decomposition</td>
<td>budget</td>
</tr>
<tr>
<td>functional decomposition</td>
<td>benchmarking</td>
</tr>
<tr>
<td>designing with multiple</td>
<td>performance analysis</td>
</tr>
<tr>
<td>decompositions</td>
<td>granularity determination</td>
</tr>
<tr>
<td>execution architecture</td>
<td>value and cost</td>
</tr>
<tr>
<td>internal interfaces</td>
<td>safety analysis</td>
</tr>
<tr>
<td>performance</td>
<td>reliability analysis</td>
</tr>
<tr>
<td>start up</td>
<td>security analysis</td>
</tr>
<tr>
<td>shutdown</td>
<td></td>
</tr>
<tr>
<td>integration plan</td>
<td></td>
</tr>
<tr>
<td>work breakdown</td>
<td></td>
</tr>
<tr>
<td>safety</td>
<td></td>
</tr>
<tr>
<td>reliability</td>
<td></td>
</tr>
<tr>
<td>security</td>
<td></td>
</tr>
</tbody>
</table>

Legend:
- **explicitly addressed**
- **addressed only implicitly**
- not addressed

Coverage based on documentation status of first product release
The case material is based on actual data, from a complex context with large commercial interests. The material is simplified to increase the accessibility, while at the same time small changes have been made to remove commercial sensitivity. Commercial sensitivity is further reduced by using relatively old data (between 5 and 10 years in the past). Care has been taken that the illustrative value is maintained.
Abstract

Story telling was not used explicit during the development of the medical imaging workstation. Two stories which did have a great impact of the development of the product are described: “The sales story” and “The radiologist at work”.

The relation of the stories to the requirements and design is shown.
Stories used during development

The sales story how to capture the interest of the radiologist for the product.

The radiologist at work describing the way a radiologist works, which explains why the radiologist is not interested in viewing, but very interested in films.

The gastro intestinal examination how the URF system is used to examine patients with gastro intestinal problems. This story is not described here, because it is outside the scope of the discussed thread of reasoning.
Main sales feature: easy viewing

Try it yourself,
see how easy it is

Yes, this is great!

ECR'91 European Congress of Radiology

salesman

radiologist
Remote control makes viewing easy

next / previous examination
next / previous image
increase / decrease contrast
increase / decrease brightness
Radiologist workspots and activities

supervision of the examination

view and diagnose, dictate report

verify and authorise report

activities of the radiologist
Diagnosis in tens of seconds

- Films loaded by clinical personnel during the day
- Light-box
- Auto-loader

- Looks at images
- Moves head forward / backward
- Moves head or eyes left/right/up/down
- Zooms in
- Overview
- Mumbles a few Latin words or clinical codes in recorder
- Presses next button

- New films
- Old films

- Report

- Image selection, panning

- Tens of seconds
From story to design

- **Customer objectives**
- **Application**
- **Functional**
- **Conceptual**
- **Realization**

**ease of use**

**sales story**

- response time, minimal UI
  - response times: image retrieve, C/B RC functionality

**film efficiency**

**radiologist at work**

- processing throughput and quality
  - memory budget
  - CPU load
  - network load
  - disk budget
  - algorithms

- analyse design

**Customer objectives**

- 20 1024 ² 8 bit images
- 3 films of 4k*5k pixels per examination
- 4 exams / room
- 3 rooms/workstation
Threads of Reasoning in the Medical Imaging Case

by Gerrit Muller    Buskerud University College

e-mail: gaudisite@gmail.com

www.gaudisite.nl

Abstract

A thread of reasoning is build up in steps and the underlying reasoning is explained.
Thread of reasoning based on efficiency-quality tension

Customer objectives
- time efficient
- diagnostic quality
- safety (liability)

Specification issues
- system response
- system throughput

Concepts
- resource management
- processor, memory
- internal logistics
- concurrency, processes
- image processing
- algorithms
- design
- space

Threads of Reasoning in the Medical Imaging Case
56   Gerrit Muller

version: 2.4
July 31, 2014
MITORthread
Thread of reasoning; introvert phase

Introvert view: cost and impact of new technologies
Thread of reasoning; phase 2

Customer objectives

Application

Functional

Conceptual

Realization

How to measure memory, how much is needed? from introvert to extrovert

Philips operational view (manufacturing, service, sales)
Radiologists diagnose from film, throughput is important. Extrovert view shows conceptual and realization gaps!
Threads of reasoning in the Medical Imaging Case

from extrovert diagnostic quality, via image quality, algorithms and load, to extrovert throughput

Philips operational view
(manufacturing, service, sales)
cost revisited in context of clinical needs and realization constraints; note: original threads are significantly simplified