From story to design illustrated by medical imaging

-

Gerrit Muller
University of South-Eastern Norway-NISE
Hasbergsvei 36 P.O. Box 235, NO-3603 Kongsberg Norway
gaudisite@gmail.com

Abstract

The medical imaging workstation is used as printserver for multiple X-ray examination rooms. A quantified story is used to understand the use of the system, and to analyse specification and design.

version: 1.0 status: concept September 6, 2020

1 Introduction

This article is a subset of the book on Architectural Reasoning [2]. Story telling is explained and illustrated by means of extensive selective *copy and paste* work from this book.

2 Story Telling

Story telling is explained in this article on the basis of the CAFCR model [2]. The "CAFCR" model is a decomposition of an architecture description in 5 views, as shown in the top of Figure 1. The *customer objectives* view (**what** does the customer want to achieve) and the *application* view (**how** does the customer realize his goals) provide the needs of the customer. The needs of the customer (**what** and **how**) provide the justification (**why**) for the specification and the design.

The *functional* view describes the **what** of the product, that includes (despite the name) also the *non functional* requirements. The **how** of the product is described in the *conceptual* and *realization* view. The **how** of the product is split into 2 separate views for reasons of stability: the conceptual view is changing less in time than the fast changing realization (Moore's law!).

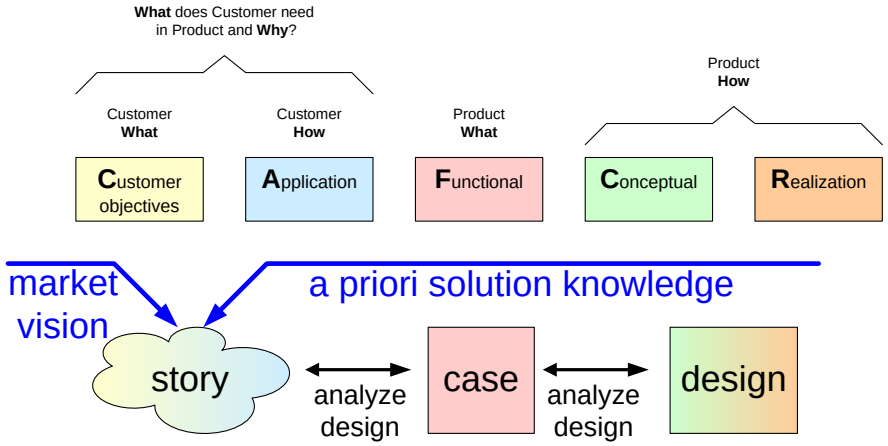

Figure 1: From story to design

The method provided here, based on story telling, is a powerful means to get the product definition quickly in a concrete factual discussion. The method is especially good in improving the communication between the different stakeholders. This communication is tuned to the stakeholders involved in the different CAFCR views: the *story* and *use case* can be exchanged in ways that are understandable for both marketing oriented people as well as for designers.

Figure 1 positions the story in the customer objectives view and application view. A good story combines a clear market vision with a priori realization know how. The story itself must be expressed entirely in customer terms, no solution jargon is allowed. The story is used to analyze specific parts of the specification: a use case. The use case is then used to explore specific parts of the design.

3 Medical Imaging Workstation

The story telling method is illustrated by means of the *Medical Imaging Workstation* case. The Easyvision is a medical imaging workstation, that provides additional printing functionality to URF X-ray systems, see Figure 2. In a radiology department three URF examination rooms can be connected to a single Easyvision. The Easyvision can process and print the images of all three URF systems on transparent film. The radiologist is viewing the film on a light box to perform the diagnosis.

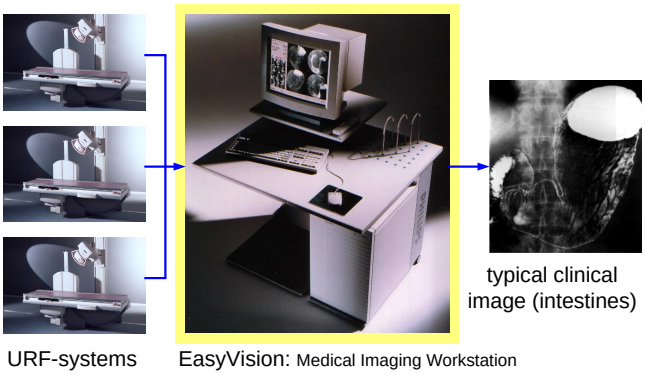

Figure 2: Easyvision serving three URF examination rooms

URF systems are used in gastrointestinal examinations. The patient has to consume barium meal to enhance the contrast. Multiple exposures are made at different locations in the intestines, while the barium meal progresses. The radiologist applies wedges to expose the area of interest and to minimize the X-ray dose for the rest of the body.

The introduction of the Easyvision makes it possible to connect 3 examination rooms via an Easyvision to a digital laserprinter. Figure 3 shows that the Easyvision can be positioned as a server in some cabinet, in that case the system is used remotely, without any direct operator interaction. The Easyvision can also be placed in one of the control rooms, that enables manual processing of the images and formatting of the film.

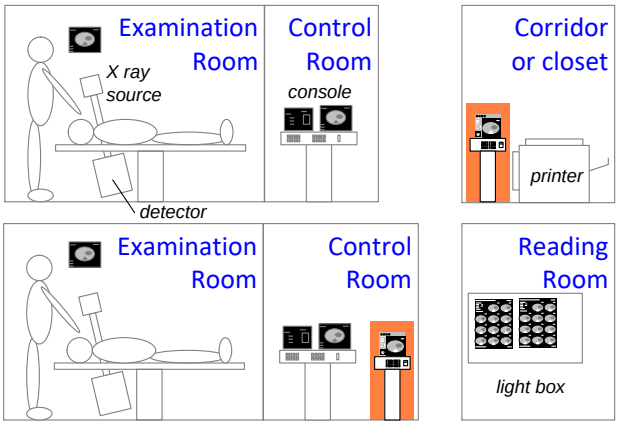

Figure 3: X-ray rooms with Easyvision applied as printserver

4 The Radiologist Story

The radiologist has the following activities, which are directly related to the diagnosis of a patient: supervising the examination, viewing the images to come to a diagnosis, dictate a report and verify and authorize the textual version of the report. Figure 4 shows these activities.

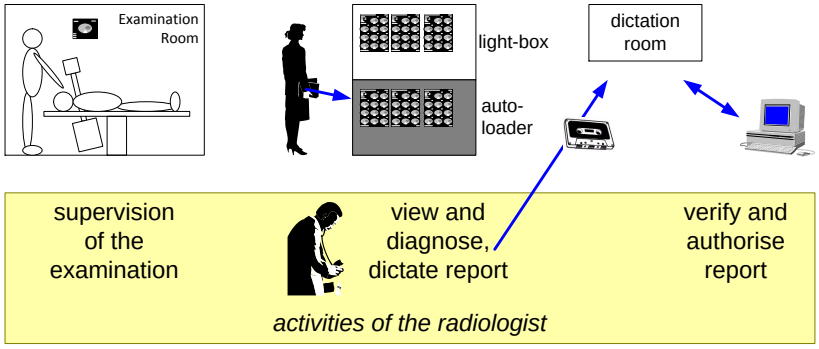

Figure 4: Radiologist work-spots and activities

The radiologist is responsible for the image acquisition in the examination room. He is not full-time present in the examination rooms, but supervises the work in multiple rooms. The radio technicians and other clinical personnel do most of the patient handling and system operation.

The films with examinations to be viewed are collected by clinical personnel and these films are attached in the right order to carriers in the auto-loader. The auto-loader is a simple mechanical device, which can lift a set of films out of the

store to the front of the lightbox. One press of the button removes the current set of films and retrieves the next set of films.

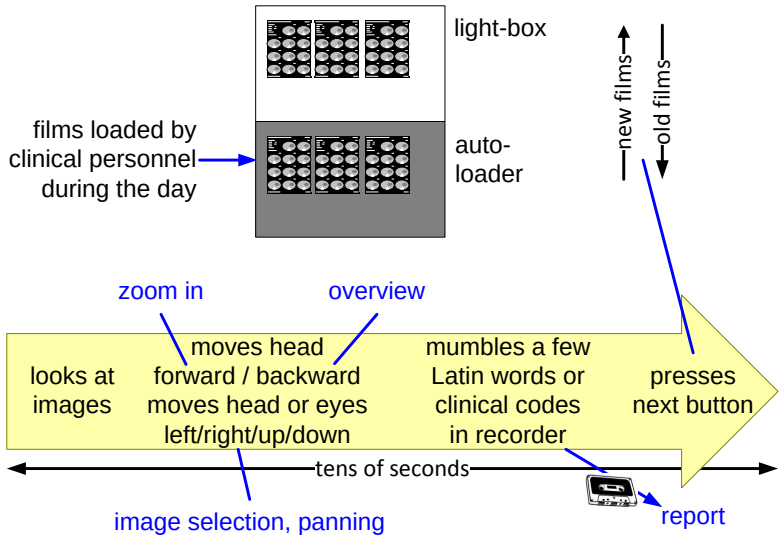

Figure 5: Diagnosis in tens of seconds

The viewing and determining the diagnosis is an amazing short during activity. Figure 5 shows this activity in some more detail. A few movements of the head and eyes are sufficient to get the overview and to zoom in on the relevant images and the relevant details. The spoken report consists of a patient identification, a few words in latin and or some medical standard codes. The recorded spoken report is send to the dictation department, the transcription will be verified later. The radiologist switches to the next examination with a single push on the next button of the auto-loader. This entire activity is finished within tens of seconds.

Later on the day the radiologist will verify and authorize the transcribed reports.

5 Analysis of the Story into a Case

The specification and design of the medical imaging workstation was based on "typical" cases. Figure 6 shows the typical case for URF examinations. Three examination rooms are sharing one medical imaging workstation. Every examination room has an average throughput of 4 patients per hour (patient examinations are interleaved, as explained below for Figure **??**).

The average image production per examination is 20 1024^2 8-bit images. These images are printed typical on 3 film sheets of 4k*5k pixels. Image quality of the film sheets is crucial, which translates in the use of bi-cubic interpolation.

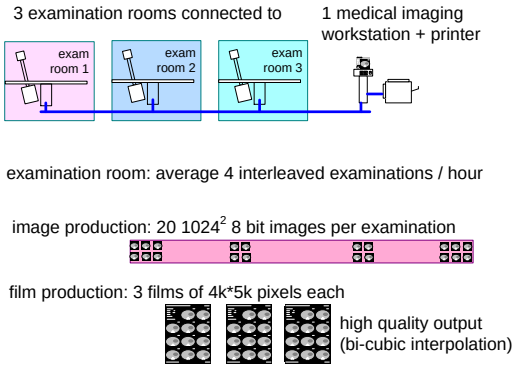

Figure 6: Typical case URF examination

6 Analysis of the Case; towards Design

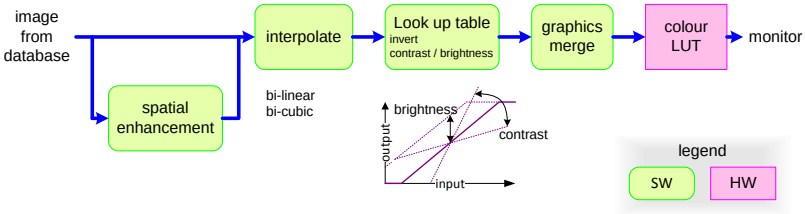

Figure 7: Presentation pipeline for X-ray images

Figure 7 shows the rendering pipeline as used in the medical imaging workstation. Enhancement is a filter operation, where the enhancement kernels are predefined in the acquisition system. The interpolation is used to resize the image from acquisition resolution to the desired viewport (or film-port) size. The grey-levels for display are determined by means of a lookup table. This lookup table normally has a linear content, where the slope determines the contrast and the vertical offset the brightness of the image. Finally graphics and text are superimposed on the image, for instance for image identification and for annotations by the user.

The image interpolation algorithm used depends on desired image quality and available processing time. Bi-linear interpolation is an interpolation with a low-pass filter side effect, the image becomes less sharp. A bi-cubic interpolation is an approximation of the more ideal sinc() based interpolation. The bi-cubic interpolation is parameterized, where the parameter settings determine how much the interpolation causes low pass or high pass filtering (blurring or sharpening). These bi-cubic parameter choices are normally not exported to the user interface, empirical values are used.

memory budget in Mbytes	code	obj data	bulk data	total
shared code	11.0			11.0
User Interface process	0.3	3.0	12.0	15.3
database server	0.3	3.2	3.0	6.5
print server	0.3	1.2	9.0	10.5
optical storage server	0.3	2.0	1.0	3.3
communication server	0.3	2.0	4.0	6.3
UNIX commands	0.3	0.2	0	0.5
compute server	0.3	0.5	6.0	6.8
system monitor	0.3	0.5	0	0.8
application SW total	13.4	12.6	35.0	61.0
UNIX Solaris 2.x				10.0
file cache				3.0
total				74.0

Figure 8: Example of a memory budget

The amount of memory in the medical imaging workstation is limited for cost reasons, but also for simple physical reasons: the workstation used at that moment did not support more than 64 MByte of physical memory. The workstation and operating system did support virtual memory, but for performance reasons this should be used sparingly.

A memory budget is used to manage the amount of memory in use. Figure 8 shows the memory budgets of release 1 and release 2 of Easyvision RF side by side. Three types of memory are distinguished: program or code (*read-only* from operating system point of view), object data (dynamic allocated data via malloc and free, *heap* based) and bulk data for large consecutive memory areas, mostly used for images.

Per process, see Figure 9, the typical amount of memory per category is specified. The memory usage of the operating system is also specified.

The concepts in Figure 8 are:

- the processes as granularity to manage memory
- shared code between processes (also called dynamic libraries)
- operating system and related memory consumption
- three types of memory (code, object data and bulk data) with different characteristics
- memory budget per process per memory type

The execution architecture of Easyvision is based on UNIX-type processes and shared libraries. Figure 9 shows the process structure of Easyvision. Most processes can be associated with a specific hardware resource, as shown in this

figure. Core of the Easyvision software architecture is the database. The database provides *fast*,*reliable*, *persistent* storage and it provides synchronization by means of active data. Synchronization and communication between processes always takes place via this database.

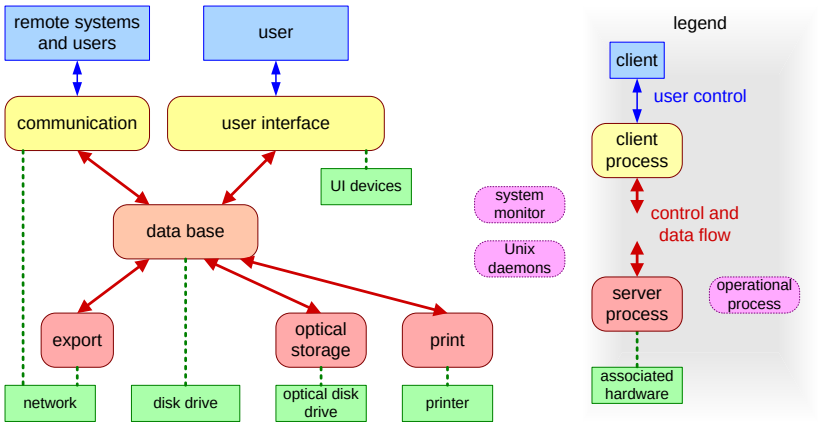

Figure 9: Software processes or tasks running concurrently in Easyvision

Figure 9 shows 2 types of processes: *client* and *server*. The client type interact with a user (remote or direct), while the servers perform their work in the background.

A process as unit of design is used for multiple reasons, ranging from managing concurrency to managing resource use. Figure **??** lists all criterions used to determine the process decomposition. One general rule is to minimize the amount of processes, because too many processes decrease the manageability, visibility and understandability.

The print server uses a different memory strategy than the user interface process, see Figure 10. The print server creates the film-image by rendering the individual images. The film size of 4k*5k images is too large to render the entire film at once in memory: 20 Mpixels, while the memory budget allows 9 Mbyte of bulkdata usage. The film image itself is already more than the provided memory budget!

The film image is build up in horizontal strokes, which are sent to the laser-printer. The size of the stroke is chosen such that input image + intermediate results + 2 bands (for double buffering) fit in the available bulkdata budget. At the same time the band should not be very small because the banding increases the overhead and some duplicate processing is sometimes needed because of edge effects.

For background tasks a CPU budget is used, expressed in CPU seconds per Mega-byte or Mega-pixel. This budget is function based: importing and printing. Most background jobs involve a single server plus interaction with the database server.

Two use cases are relevant: interactive viewing, with background jobs and pure

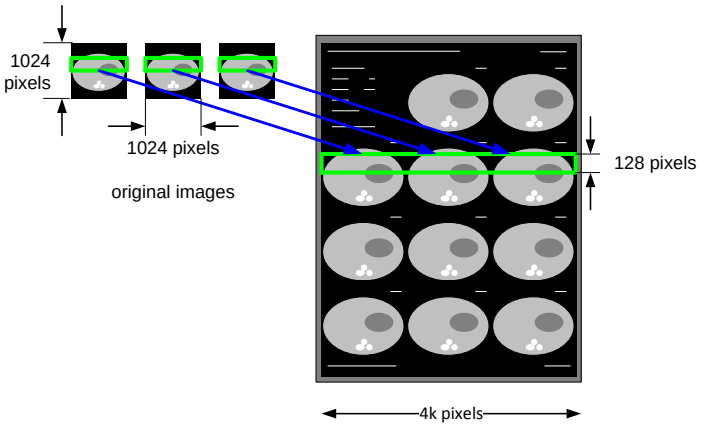

Figure 10: Print server is based on banding

print serving. For interactive response circa 70% of CPU time should be available, while the load of a full throughput case must stay below 90% of the available CPU time. Figure 11 shows both use cases.

Figure 12 shows an overview of the specification concepts, design concepts and the dimensioning of the implementation that can be done on the basis of the "Radiologist at work" story.

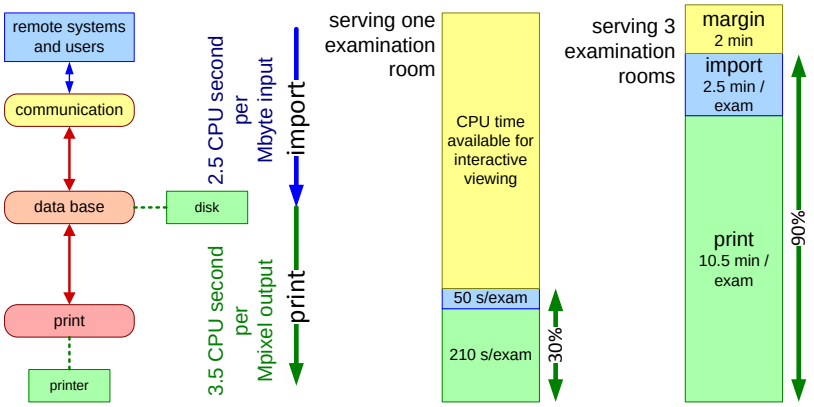

Figure 11: Server CPU load

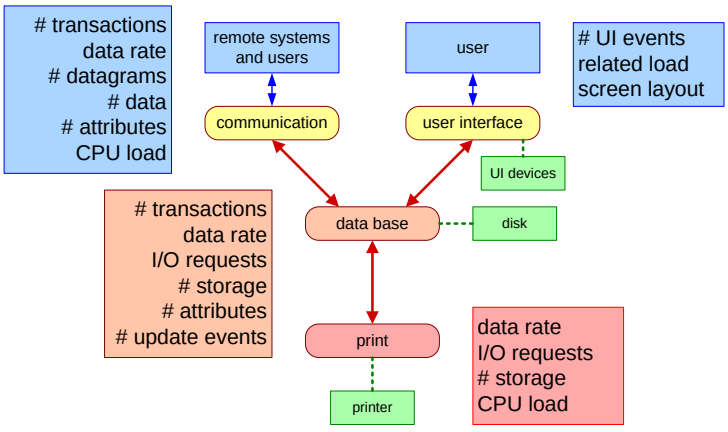

Figure 12: Analysis in realization view

7 Assessment of a story

Figure 13 shows the criterions for a good story. It is recommended to assess a story against this checklist and either improve the story such that it meets all the criterions or reject the story.

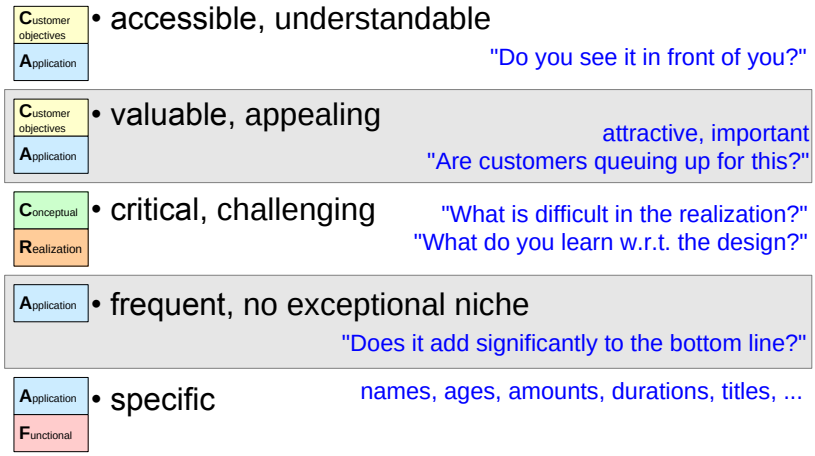

Figure 13: Criterions for a good story

Subsections 7.1 to 7.5 describe every criterion in more detail.

7.1 Accessible, understandable

The main function of a story is to make the opportunity or problem communicable with all the stakeholders. This means that the story must be accessible and understandable for all stakeholders. The description or presentation should be such that all stakeholders can *live through*, *experience* or *imagine* the story. A "good" story is not a sheet of paper, it is a living story.

7.2 Important, valuable, appealing, attractive

The opportunity or problem (idea, product, function, or feature) must be significant for the target customers. This means that it should be important for them, or valuable; it should be appealing and attractive.

Most stories fail on this criterium. Some so-so opportunity (whistle and bell-type) is used, where nobody gets really enthusiastic. If this is the case more creativity is required to change the story to a useful level of importance.

7.3 Critical, challenging

The purpose of the story is to learn, define, and analyze new products or features. If the implementation of a story is trivial, nothing will be learned. If all other criterions are met and no product exists yet, then just do it, because it is clearly a quick win!

If the implementation is challenging, then the story is a good vehicle to study the trade-offs and choices to be made.

7.4 Frequent, no exceptional niche

Especially in the early exploration it is important to focus on the main line, the *typical* case. Later in the system design more specialized cases will be needed to analyze for instance more exceptional worst case situations.

A *typical* case is characterized by being frequent, it should not be an exceptional niche.

7.5 Specific

The value of a story is the specificity. Most system descriptions are very generic and therefore very powerful, but at the same time very non-specific. A good story provides focus on a single story, one occasion only. In other words, the thread of the story should be very specific.

A common pitfall for story writers is to show all possibilities in one story. For example one paragraph that describes all the potential goodies. Simply leave out such a paragraph, it only degrades the focus and value of the story.

A good story is in **all** aspects as specific as possible, which means that:

- persons playing a role in the story preferably have a name, age, and other relevant attributes
- the time and location are specific (if relevant)
- the content is specific (for instance is listening for **2 hours** to songs of **the Beatles**)

This kind of specific data is often needed to assess the other criterions, to bring it more alive, and in further analysis. If during the use of the story numbers have to be "invented", it is recommended to improve the story by adding specific facts to the story.

References

[1] Gerrit Muller. The system architecture homepage. `http://www.gaudisite.nl/index.html`, 1999.

[2] Gerrit Muller. Architectural reasoning explained. http://www.gaudisite.nl/ArchitecturalReasoningBook.pdf, 2002.

History

Version: 1.0, date: July 23, 2003 changed by: Gerrit Muller

- Created the article version by a lot of copy paste work
- changed status to concept

Version: 0.1, date: July 23, 2003 changed by: Gerrit Muller

- Added LAC logo

Version: 0, date: July 14, 2003 changed by: Gerrit Muller

- Created, no changelog yet