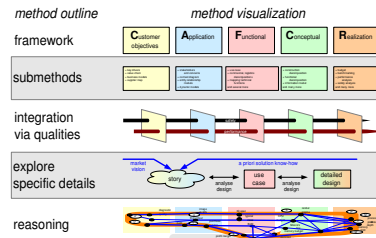


# Overview of CAFCR and Threads of Reasoning

-



Gerrit Muller

Embedded Systems Institute

Den Dolech 2 (Laplace Building 0.10) P.O. Box 513, 5600 MB Eindhoven The Netherlands

[gerrit.muller@embeddedsystems.nl](mailto:gerrit.muller@embeddedsystems.nl)

## Abstract

The described architecting method uses the CAFCR model as starting point. Qualities are used as orthogonal dimension to integrate the CAFCR views. Story telling is used to add specifics. Threads of reasoning combine all the information into a coherent overview.

### Distribution

This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution is allowed as long as the document remains complete and unchanged.

All Gaudí documents are available at:  
<http://www.gaudisite.nl/>

version: 1.5

status: finished

February 10, 2011

# 1 Introduction

At the beginning of the creation of a new product the problem is often ill-defined and only some ideas exist about potential solutions. The architecting effort must change this situation in the course of the project into a well articulated and structured understanding of both the problem and its potential solutions. Figure 1 shows that basic methods and an architecting method enable this architecting effort.

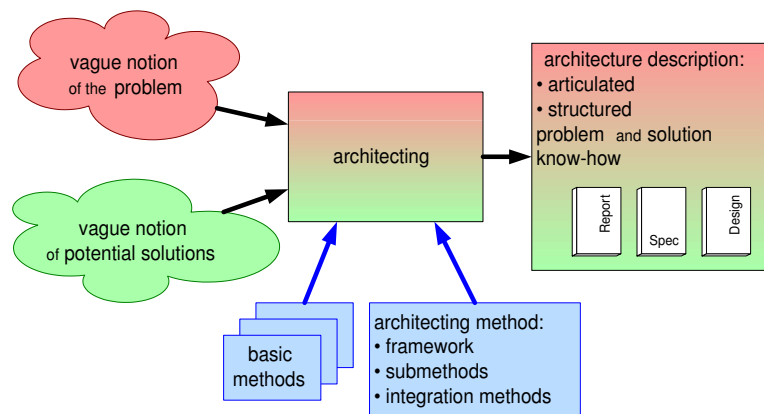


Figure 1: An architecting method supports the architect in his process to go from a vague notion of the problem and a vague notion of the potential solutions to a well articulated and structured architecture description

The basic methods are methods that are found in a wide range of disciplines, for example to analyze, to communicate, and to solve problems. These basic methods are discussed in Chapter ??.

An overview of the architecting method is given in Section 2. The architecting method contains multiple elements: a framework, briefly introduced in Section 3, and submethods and integrating methods, which are described in part II.

## 2 Architecting Method Overview

Figure 2 shows the overall outline of the architecting method. The right hand side shows the visualization as it will be used in the later chapters. The *framework* is a decomposition into five views, the “CAFRCR” model, see Section 3.

Per view in the decomposition a collection of *submethods* is given. The collections of submethods are open-ended. The collection is filled by borrowing relevant methods from many disciplines.

A decomposition in itself is not useful without the complementing integration. *Qualities* are used as *integrating* elements. The decomposition into qualities is orthogonal to the “CAFRCR” model.

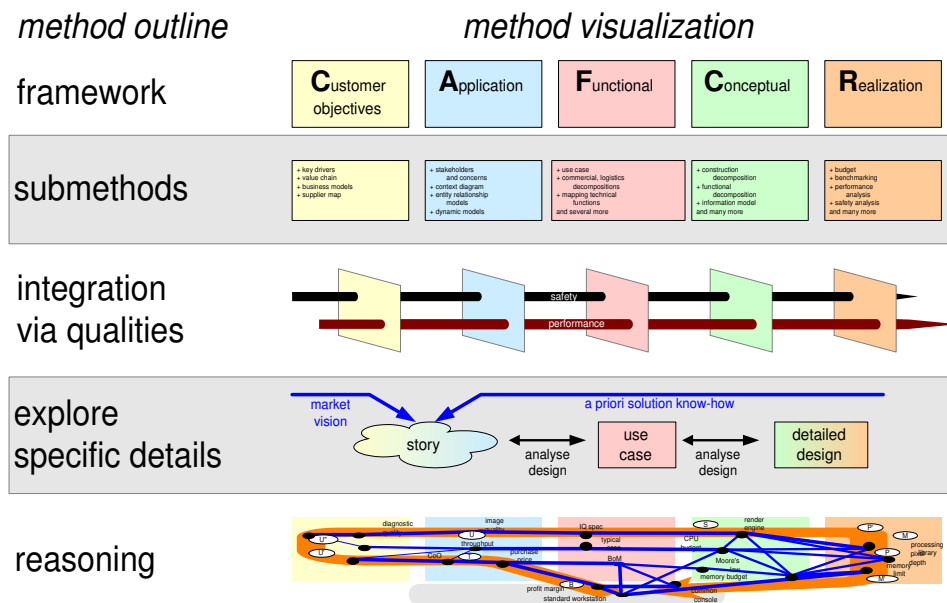


Figure 2: The outline of the architecting method with the corresponding visualization that will be used in the later chapters.

The decomposition into CAFCR views and into qualities both tend to be rather *abstract, high level* or *generic*. Therefore, a complementary approach is added to *explore specific details*: story telling. Story telling is the starting point for specific case analysis and design studies.

These approaches are combined into a thread of *reasoning*: valuable insights in the different views in relation to each other. The basic working methods of the architect and the decompositions should help the architect to maintain the overview and to prevent drowning in the tremendous amount of data and relationships. The stories and detailed case and design studies should help to keep the insights factual.

### 3 The CAFCR Model

The “CAFCR” model is a decomposition of an architecture description into five views, as shown in Figure 3. The *customer objectives* view (**what** does the customer want to achieve) and the *application* view (**how** does the customer realize his goals) capture 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, which includes (despite its name) the *non-functional* requirements.

The **how** of the product is described in the *conceptual* and *realization* views.

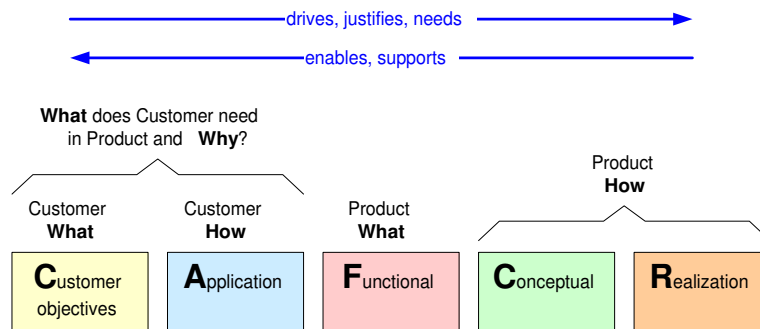


Figure 3: The “CAFCR” model

The **how** of the product is split into two separate views for reasons of stability: the conceptual view is maintained over a longer time period than the fast changing realization (Moore’s law!).

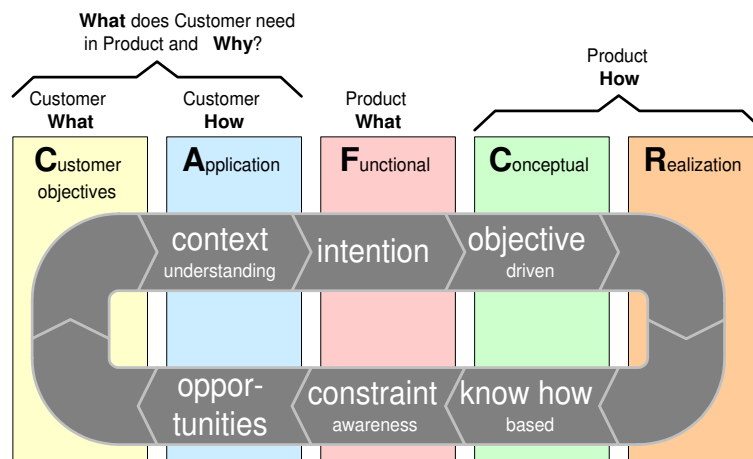


Figure 4: Iteration over the CAFCR views and the operational view. The task of the architect is to integrate all these viewpoints, in order to get a *valuable, usable* and *feasible* product.

The job of the architect is to integrate these views in a consistent and balanced way, in order to get a *valuable, usable* and *feasible* product. Architects do this job by continuously iterating over many different viewpoints, sampling the problem and solution space in order to build up an understanding of the business. This is a top-down approach (objective driven, based on intention and context understanding) in combination with a bottom-up approach (constraint aware, identifying opportunities, know-how based), see Figure 4.

The CAFCR model in Figure 4 is focused on the relation between the customer

world and the product. Another dimension that plays a role in specification and design is the *operational* view. The operational view describes the internal requirements of the company: what is needed for the operation of the company? The CAFCR model is focused on the customer world: what determines *value* and *usability* of a product? The business *feasibility* of a product is largely determined by the operation of the company: satisfactory margins, service levels, potential for the future. Strategic requirements of the company, which are important for the long term operation, are also part of the *operational* view.

The customer views and operational view are asymmetric. The customer world is outside the scope of control of the company. Customers have a free will, but act in a complex environment with legislation, culture, competition, and their own customers, who determine their freedom of choices. The operational way of working of a company is inside the scope of control of the company. The company is also constrained by many external factors. Within these constraints, however, the company decides itself how and where to manufacture, to sell, and to provide service. The operation of the company is organized in such a way that it supports its customers. The asymmetry is that a company will never tell its customers to organize in a way that eases the operation of the company<sup>1</sup>. The operational view is subject to the customer views.

The CAFCR views and the operational view must be used concurrently, not top down as in the waterfall model. However, at the end of the architecting job a consistent description must be available, see [2]. The *justification* and the *needs* are expressed in the Customer Objectives View, the Application View, and the operational view. The technical solution as expressed in the Conceptual View and the Realization View *supports* the customer to achieve his objectives and support the company in the operation. The Functional View is the interface between problem and solution world.

The CAFCR model will be used in this thesis as a framework for a next level of submethods. Although the five views are presented here as sharp disjunct views, many subsequent models and methods don't fit entirely into one single view. This in itself is not a problem; the model is a means to build up understanding, it is not a goal in itself.

The "CAFCR" model can be used recursively: many customers are part of a longer value chain and deliver products to customers themselves. Understanding of the customer's customer improves the understanding of the requirements.

The notion of **the** customer is misleading. Many products have an extensive set of stakeholders in the customer domain. One category of customer stakeholders are decision makers such as: CEO (Chief Executive Officer), CFO (Chief Financial

---

<sup>1</sup>In practice it is less black and white. A company interacts with its customers to find a mutual beneficial way of working. Nevertheless, the provider-customer relationship is asymmetric. If the provider dictates the way of working of the customer then something unhealthy is happening. Examples of unhealthy relations can be found in companies with a monopoly position.

Officer), CIO (Chief Information Officer), CMO (Chief Marketing Officer) and CTO (Chief Technology Officer). Another category are people actually operating the system, such as users, operators, and maintainers. A last category mentioned here are the more remotely involved stakeholders, such as department chiefs and purchasers.

## References

- [1] Gerrit Muller. The system architecture homepage. <http://www.gaudisite.nl/index.html>, 1999.
- [2] D.L. Parnas and P.C. Clements. A rational design process: How and why to fake it. *IEEE Transactions on Software Engineering*, SE-12., No. 2:251–257, February 1986.

## History

**Version: 1.5, date: April 4, 2004 changed by: Gerrit Muller**

- updated overview figure CAFCR: removed operational view
- updated overview of complete method: other representation of operational view
- text changed accordingly

**OldVersion1.4April 1, 2004Gerrit Muller**

- changed title into Overview of CAFCR and Threads of Reasoning

**Version: 1.3, date: March 17, 2004 changed by: Gerrit Muller**

- changed status to finished

**Version: 1.2, date: November 24, 2003 changed by: Gerrit Muller**

- Changed title into "Overview of CAFCR and Architectural Reasoning"
- many small textual improvements
- changed status to draft

**Version: 1.1, date: July 31, 2003 changed by: Gerrit Muller**

- Added operational view to figure and explanation of operational view
- changed status to preliminary draft

**Version: 1, date: July 28, 2003 changed by: Gerrit Muller**

- Added overview of the CAFCR model

**Version: 0, date: May 20, 2003 changed by: Gerrit Muller**

- Created, no changelog yet