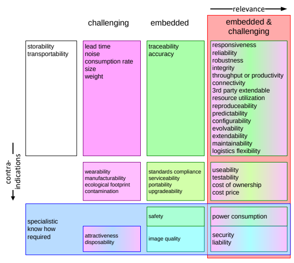
Capability development at the Embedded Systems Institute

-



Gerrit Muller
Embedded Systems Institute
Hasbergsvei 36 P.O. Box 235, NO-3603 Kongsberg Norway
gaudisite@gmail.com

Abstract

The *systems* discipline is decomposed in views and qualities and complemented with a framework to integrate again. The qualities are taken as starting point to define system design capabilities. These capabilities are analyzed and a set of embedded system capabilities is proposed.
The ESI approach with projects and capabilities is described. The contribution of ESI is explained. Some background is provided about the technology management and research method aspects.

version: 1.0 status: draft September 1, 2020

1 Introduction

The objective of embedded systems institute is to build up, consolidate and transfer knowledge how to create embedded systems effectively. Figure 1 summarizes the role of the institute by means of an annotated sentence.

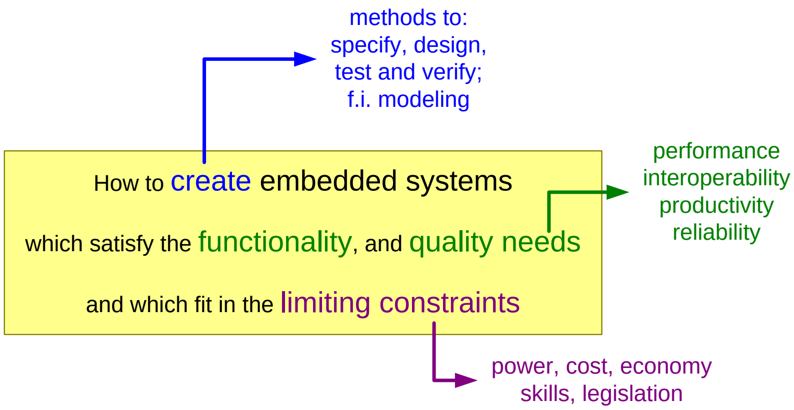


Figure 1: Role of Embedded Systems Institute ESI

To create embedded systems means and methods are needed to specify, design, test and verify, for instance modelling methods. These embedded systems must fulfil the functionality and the quality needs. The encompassing method framework is described in 2 and the capabilities in terms of qualities are described in section 3.

Embedded systems are mostly created in an industrial setting, which means that many practical constraints are present during the product creation. The creation methods must cope in a practical way with these constraints. The approach taken by the Embedded Systems Institute, a close cooperation with an industrial partner, is described in section4. Section 5 describes the contribution of the Embedded Systems Institute.

Section 6 is copied from [3] and describes the relation between technology management and method research.

2 Method framework

A useful top level decomposition of an architecture is provided by the so-called "CAFCR" model, as shown in figure 2. The *customer objectives* view and the *application* view provide the **why** from the customer. The *functional* view describes the **what** of the product, which includes (despite the name) also the *non functional* requirements.

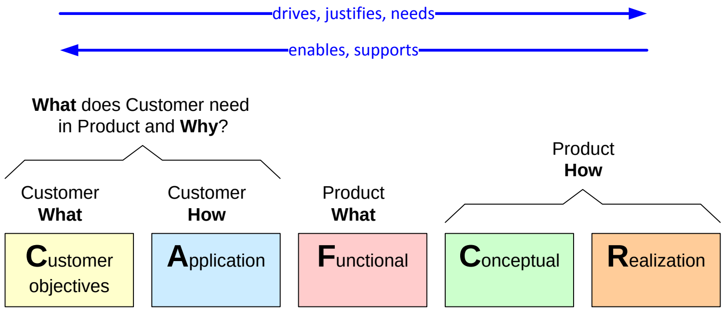


Figure 2: The "CAFCR" model

The **how** of the product is described in the *conceptual* and *realization* view, where the conceptual view is changing less in time than the fast changing realization (Moore's law!).

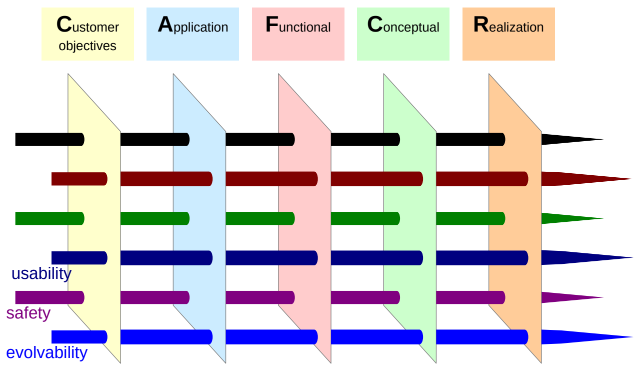


Figure 3: Qualities as basis for capabilities

The 5 CAFCR views become more useful when the information in one view is used in relation with neighboring views. One of the starting points is the use of the stakeholder concerns. Many stakeholder concerns are abstracted in a large set of more generic qualities. These qualities are meaningful in every view in their own way. Figure 3 shows the qualities as cross cutting needles through the CAFCR views.

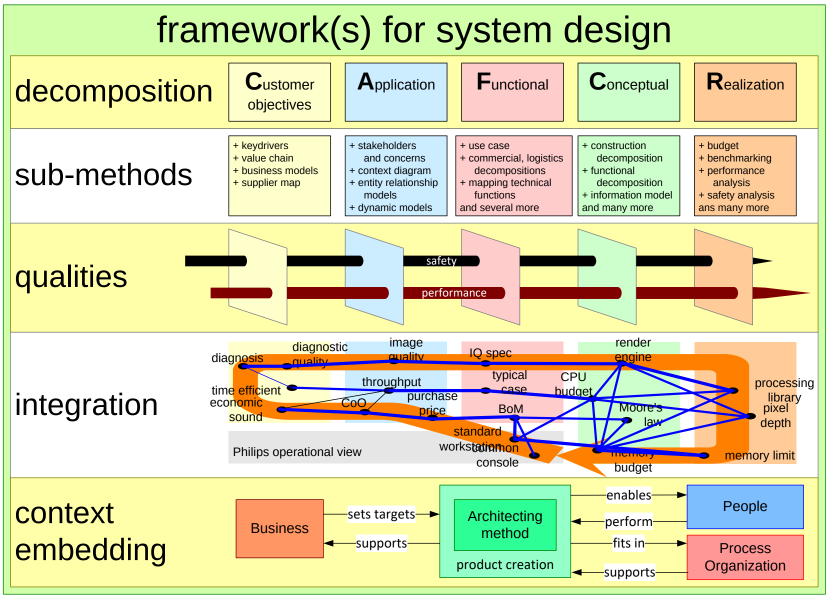


Figure 4: Overview of methods in relation with context

The CAFCR model and the qualities are the 2 main decomposition directions for a method framework, as shown in figure 4. Many submethods play a role in one view or along one of the qualities. Note that every decomposition requires the complementary integration. A method which helps to design for safety must be complemented with a method to design a system which fulfills all the required qualities.

The methods need to be used in an industrial setting, with its business objectives (profit, return on investment, market share, et cetera) and with existing people and processes. Of course there exists a mutual interaction between architecting method and this context, the context is not frozen.

3 System qualities as starting point for capabilities

In [4] a checklist of qualities is described, with the qualities shown in figure 5.

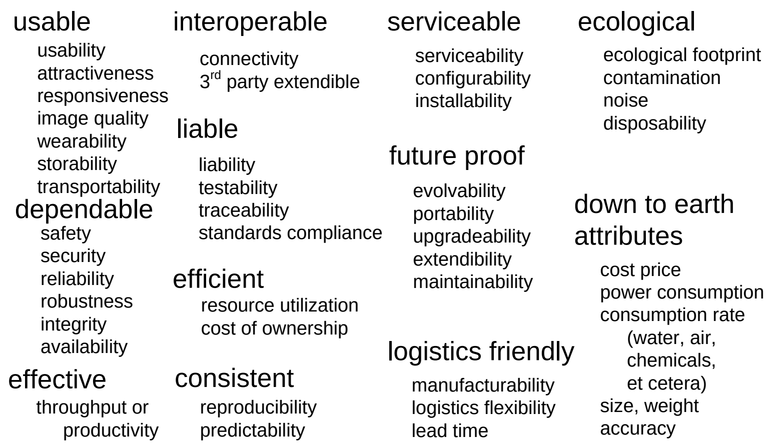


Figure 5: Checklist of system qualities

Too many qualities exist to do research all of these qualities at the same time. The qualities can be ranked with respect to their relevance for embedded systems research of system design methods. Figure 6 shows an example of such a ranking for the domain dependency of the qualities.

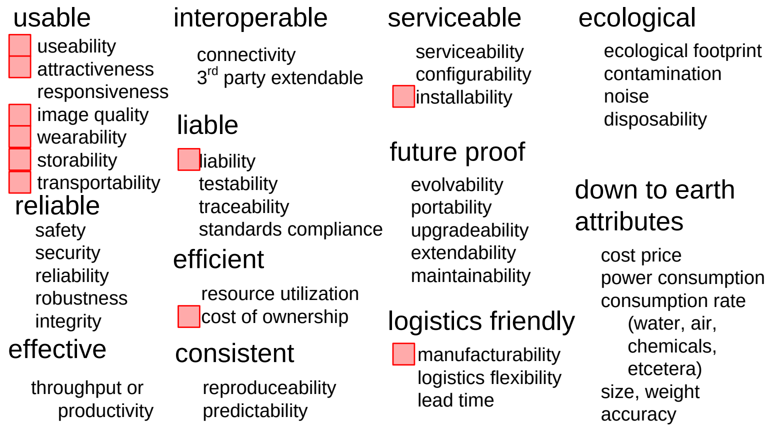


Figure 6: Domain specific aspects

The most interesting qualities for ESI are:

- related to software intensive and electronics
- are challenging
- not too domain specific
- not too much determined by process, organization or other soft factors
- not too specialistic

These criterions are shown in figure 7, together with the desired profile.

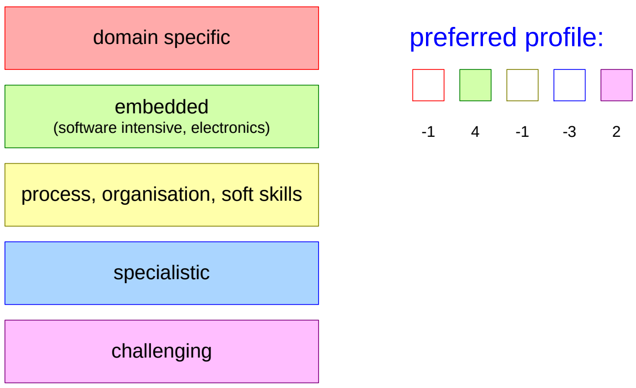


Figure 7: Preferred profile for ESI capabilities

The color codes as shown in figure 7 are used to score all qualities against these criterions, see figure 8.

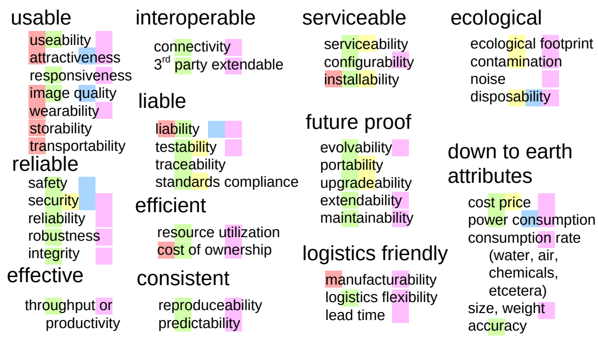


Figure 8: Ranking of all criterions

In figure 9 the qualities are classified by means of these criterions. It will be clear that qualities in the column challenging and related to embedded systems are the potential capabilities. This column is in vertical direction divided in 2 parts: qualities which are highly specialistic and the broader qualities. At this level a further vertical difference is made for the amount of contra-indicators: at the top no contra indicators, at the bottom the qualities with contra indicators.

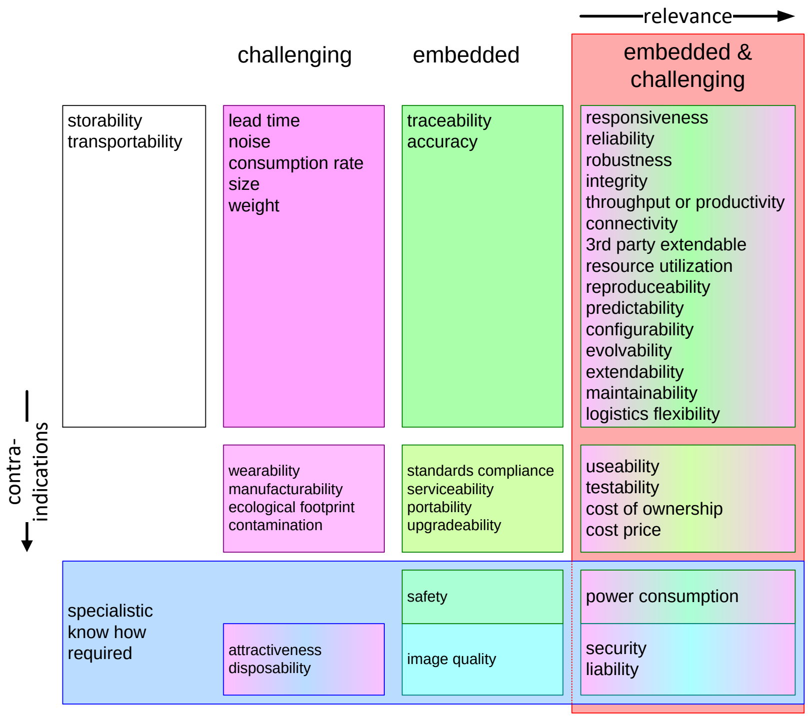


Figure 9: Relevance for ESI quality map

The broader qualities without contra-indicators are clear target capabilities. The presence of contra-indicators requires a more careful consideration of the capability.

The specialistic system design capabilities should only be targeted if a strong complementary partner is identified.

4 Projects and capabilities

Projects are used as carrier to develop the desired capabilities. Figure 10 shows the relation and the objectives of projects and capabilities. A project is done in an industrial setting and addresses an actual and important industrial problem. In this particular domain this problem is solved.

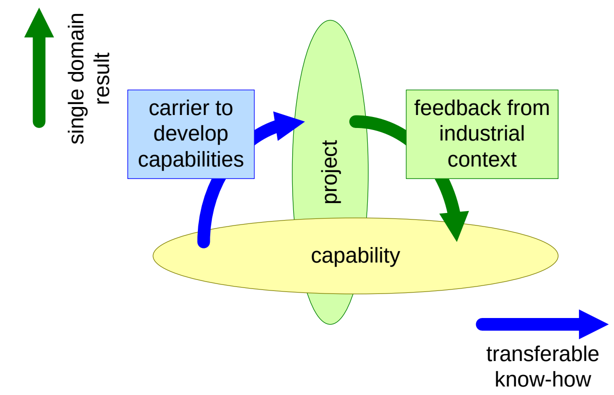


Figure 10: Project as carrier for capability development

The Embedded Systems Institute has the objective to solve the problem more generic and to transform the lessons learned in transferable know-how. The objective of the institute is capability oriented.

The projects running in an industrial environment provide a realistic context for the capability development and generate a lot of feedback for the capability development.

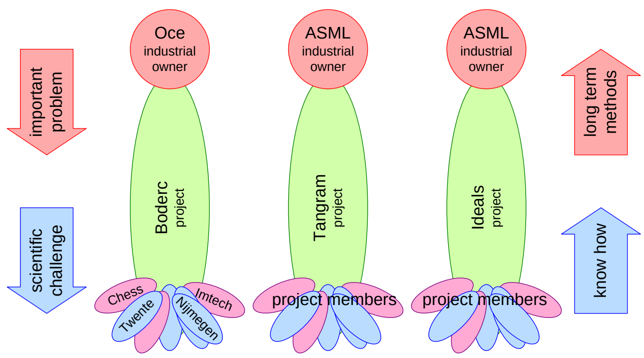


Figure 11: ESI project approach

Figure 11 show the typical structure of ESI projects. An industrial carrying partner formulates an actual and important system creation problem. The problem is anchored by means of a clear owner in the industrial partner. The project members come from many different sources of know-how. The project must offer sufficient scientific challenge to make the project attractive for academic partners.

The know how of the participants should result in a solution methods which will help the industrial partner to cope with these problems in future projects. From industrial perspective this is a long term (2 to 4 years) investment.

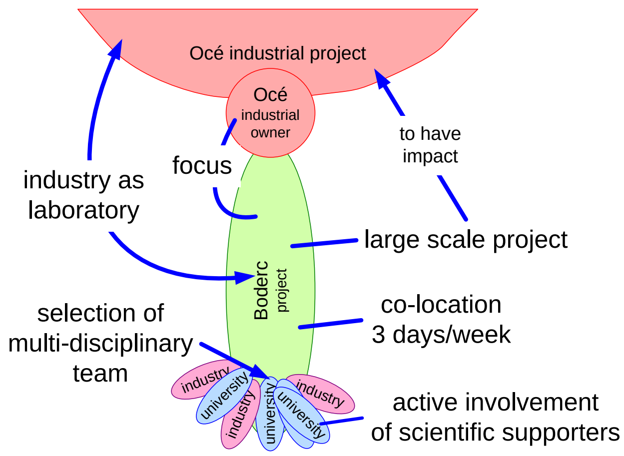


Figure 12: Critical Success Factors for projects

The projects are defined in such a way that the system design aspects can be studied:

- multi-disciplinary project members (many system problems occur at discipline boundaries)
- industry as laboratory: close cooperation of industry and research project. The size and rhythm of industrial projects are the next source of system problems. By using the industry as playground for the research true feedback can be obtained.
- focus by a clear industrial ownership to prevent local hobby-horses.
- active involvement of the scientific supporters (professors, coaches) to ensure a consistent coaching of the full time researchers.
- co-location of all team members for 3 days per week. During these days the interaction between them is stimulated.
- a substantial project size to have impact at the industrial partner.

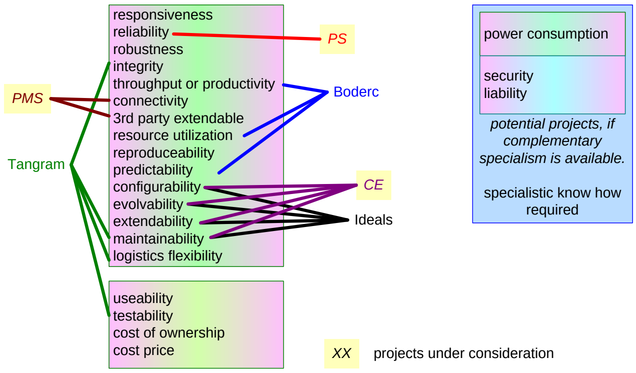


Figure 13: Mapping of capabilities to projects

The capabilities can be mapped on the projects, as shown in figure 13 Ideally capabilities are addressed in different domains, which helps to find and verify the more generic methods.

5 Role of the Embedded Systems Institute

The embedded systems institute contributes to both the project as well as to the capability development. The contribution can be decomposed in *project*, *capability* and facilitation contribution, as shown in figure 14.

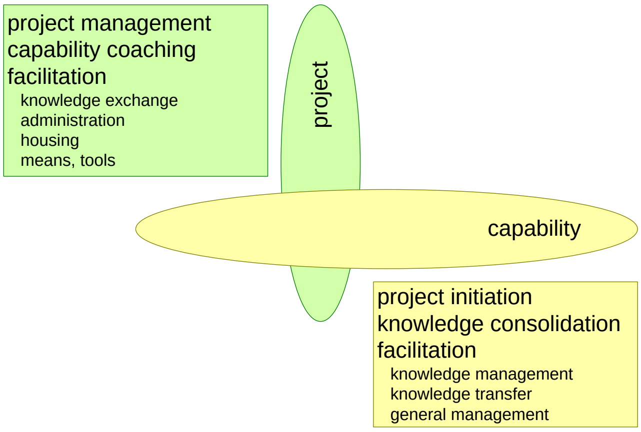


Figure 14: Role of Embedded Systems Institute 2

The ESI performs the project management of the projects itself, ensuring a smooth operation of the **cooperating** project and enabling the members to focus on the problem itself. The ESI also facilitates the knowledge exchange in the projects and the practical aspects such as administration, housing and means and tools.

The capability coaching with respect to the typical systems capabilities is performed by the ESI research fellows. The research fellows must help the team members to bridge the distance between multi-disciplinary industrial problem statement and the many single disciplinary know-how sources.

ESI plays also a role in setting up projects. Close contacts with the industry are required to know and recognize the actual and important industrial problems. Close contacts with universities and academics are required to select a multi-disciplinary team. Contacts with financial sponsors are required to fit the projects in the financial infra-structure.

The ESI has a general facilitating function, to organize the knowledge management and transfer. Consolidation of the build up capabilities is crucial, otherwise the know-how is volatile. The consolidation is done by the research fellows.

6 Technology Management Cycle

Technology management can be modeled as a cyclical process [1], as shown in figure 15. Most of the time is spent in the application of technology, in other words in the creation of new systems. After applying the technology it is recommended to learn from this application by reflection. The learning experience can (partially) be made accessible to others by consolidating the know how, for instance in documentation.

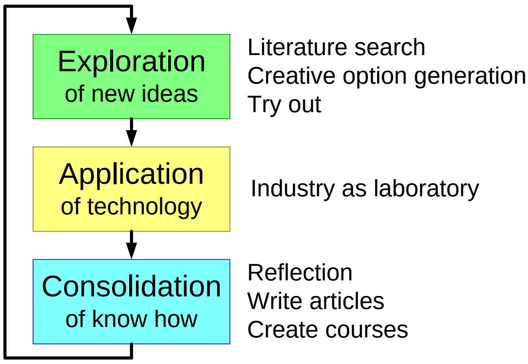


Figure 15: Technology Management Cycle

At the end of the consolidation insight will exist in strengths and weaknesses of the technology, both in the hard technology choices as well as in the soft technology (the approach taken). It is recommended to take this know how as a starting point for an exploration phase.

The exploration phase should be used to refresh the designers and architects and open new opportunities in technology. This requires that they know the state of the art in the world, by reading literature, visiting conferences et cetera. Via creative brainstorms new technology options can be added. Promising technology must be explored hands-on.

In the next application phase a limited set of new technologies is applied in practice.

Note that most effort in technology management is spent on core (hard) technologies. Hard technology is based on know how from the sciences: mathematics, physics, chemistry, biology. The know how from these sciences is very objective and universally applicable (the elasticity in the USA is the same as the elasticity in China).

A small amount of the effort will be spent on the methods required to apply this technology successfully, the methods or soft technologies. This is shown in figure 16 by the slightly darker right hand sight of the technology management cycle. Soft technologies are based on a mixture of sciences and human arts. The know how of soft technologies is more subjective, the human factors are less well reproducible (a method working well in the USA might fail in China and vice versa).

A method is by definition based on sciences and human arts: a method is a way of working for humans to use the hard technology effectively.

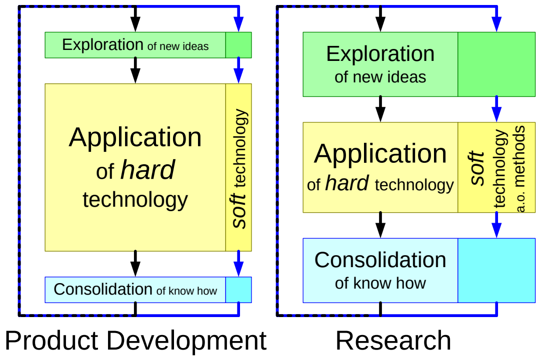


Figure 16: From Product Division to Research

This technology management cycle can be applied at multiple levels: from a design team of a specific product, up to the CTO office of a large multi-national like Philips. Design teams in the business lines will normally spend only a limited amount of time for consolidation and exploration (business pressure creates a large degree of pragmatism).

Research departments, with the explicit task of creating technology options, can spend more time on exploration and consolidation, see figure 16. However for research departments application of the technology is more difficult, this might cost a lot of time and energy, while the application might still not be realistic. Hard specific technology is more easily applied in research environment than the soft technology as architecting methods. Architecting methods are inherently related to the problems of large design teams, with all kinds of fuzzy context constraints. For that reason research of architecting methods makes use of *the industry as laboratory*: a close cooperation of research with regular design teams, where research options are tried out in a real world context.

Most effort in technology management is spent on the hard technology (which generates more direct value, for instance via Intellectual Property), while sufficient effort should be spend on methods to apply these technologies in creating new systems. In research groups with a specific capability in soft technologies the balance between hard and soft technology can be shifted somewhat more to the soft technology. To prevent that such a group floats away in abstractions sufficient hard technology should be researched at the same time. Figure 16 shows this shift in balance from hard to soft technology as well.

The move from product creation to management of architecting methods to

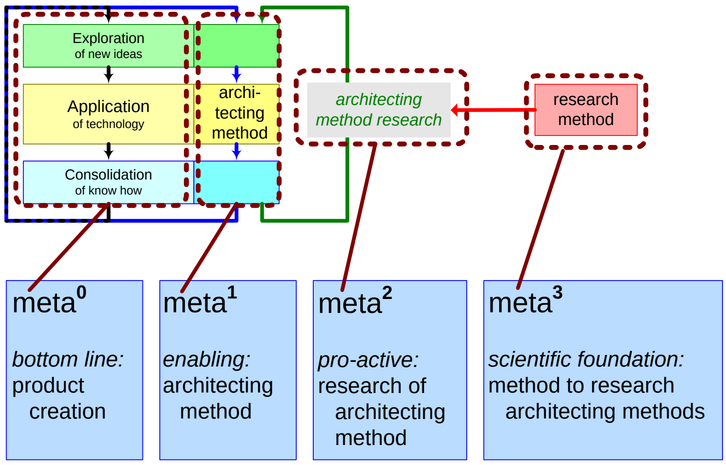


Figure 17: Moving in the *meta* direction. Research of architecting methods is two steps of indirection away from the bottomline of product creation. The scientific foundation for this work is another indirection step

research of architecting methods is a move in which the abstraction level is increasing. It is a move in the *meta* direction, as shown in figure 17.

References

[1] Hay Management Consultants. Technology management cycle. Hay Managament Consultants showed me this model in 1997/1998, taken from an article by a Japanese author. The original title of the Japanese article is unknown.

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

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

[4] Gerrit Muller. Qualities as integrating needles. `http://www.gaudisite.nl/QualityNeedlesPaper.pdf`, 2002.

History

Version: 1.0, date: June 10, 2003 changed by: Gerrit Muller

- added role and context overview
- added technology management
- added abstract
- changed status to draft

Version: 0, date: June 6, 2003 changed by: Gerrit Muller

- Created, no changelog yet