

How Reference Architectures support the evolution of Product Families¹

Gerrit Muller

Embedded Systems Institute
P.O. Box 513, 5600 MB Eindhoven
gerrit.muller@esi.nl

Introduction

Products in product families are often derived from a common platform. The platform is a means to re-use past knowledge and efforts. Many successful platforms exist, from cell phones and televisions [Ommering 2000], to cars and airplanes. However, most of the platforms are successful in more mature markets, where change is mostly extension and (cost) optimization. Platform strategies struggle more in dynamic environments where continuously many changes are imposed on the products. When the organization size also grows beyond a critical mass, then the platform strategy hits the next obstacle: (over) specialization of project members and practical limits of communication between all project members.

Our hypothesis is that Reference Architectures improve the capability to evolve platform based product families when the organization size has exceeded the critical boundary imposed by interpersonal communication. Reference Architectures start to capture the essence of Business and Technical Architectures of past systems. Based on strategy, vision and future stakeholder needs, Reference Architectures are adapted to facilitate future products. Our hypothesis is based on the communication

value of Reference Architectures for stakeholders. A compact set of artefacts that together form a Reference Architecture can more easily be shared by large teams that create platforms and products. Reference Architectures make relationships and key design decision explicit, which supports the reasoning about high-impact changes.

Industry-as-laboratory

We research Evolvability by means of the industry-as-laboratory [Potts 1993] approach in the Darwin project. Our laboratory is the development department of Magnetic Resonance Imaging (MRI) scanners. Since 1981 Philips Health Care develops and sells MRI scanners. In these decades a broad successful family of products has grown with many clinical applications, see [Laar 2007].

The clinical market of MRI scanner is still evolving fast. Trends in health care require more integration in the hospital work flow. Technical innovations enable new applications: base technology changes, such as computing and communication, are pushed forward by telecom and PC industry and are forced upon the MRI scanner engineers. Mergers and take-overs have grown the development into a large multi-site organization.

¹

This work has been carried out as part of the Darwin project under the responsibility of the Embedded Systems Institute. This project is partially supported by the Netherlands Ministry of Economic Affairs under the BSIK program.



Figure 1. Achieva MRI scanner. (Photo courtesy Philips Health Care)

Our research team explores the MRI product family in several ways:

- analyzing dependencies in the multi-million line software repository
- analyzing future internal communication infrastructure needs
- analyzing *past* use cases, where change was difficult
- analyzing change cases to study the impact of currently *anticipated* changes

The results have to be transformed in contributions to the MRI Reference Architecture. By close cooperation with MRI engineers we have to evaluate the impact of the availability of (partial) Reference Architectures on the effectiveness and efficiency of evolving the MRI system architecture, design and realization.

Problem Analysis

In professional industries such as health care the following high level problems occur:

- high effort and cost of installed base management; large diversity of system configurations
- development efficiency too low; consequence too costly too much effort required and products too late
- innovation rate too low, innovations too late

Underlying root cause is that changes that appear to be local have in practice

dependencies throughout the system. The challenge is how to apply changes locally to explore potential value and feasibility challenges. For MRI scanners the properties patient throughput, system responsiveness, image quality, safety, and reliability must be maintained. It is much easier to make a change when these properties can be sacrificed. To get sufficient clinical feedback about the value of the change these properties are crucial.

Solution direction

In discussions with the system architects we formulated the following postulates:

1. A system architecture that supports this level of exploration also supports the next phases of innovation: scaling-up and engineering.
2. A system architecture that supports this level of exploration also supports life cycle business over many generations.

The reasoning is that in the early phases the uncertainties are in the clinical needs and technical feasibility. Once these are clear then engineering aspects, such as cost, manufacturing and service time and cost get much more important. However, these can be addressed with a more rigorous engineering process, because the uncertainties have been reduced. The interfaces that are needed for exploration can be designed to be expansion interfaces for later life cycle business.

In fact the postulates are a reincarnation of old architecting wisdom: decompose and manage interfaces, separate concerns, minimize coupling, strive for cohesion, et cetera [Parnas 1972, Parnas 1978]

Analysis of evolvability problem

Based on the high level problem statement that we have formulated we want to research the evolvability property of the MRI product family. Evolvability is the capability to change in (small) steps to adapt to new changing circumstances. Evolvability is a hot research

topic; see for instance also [Fricke 2005, Ring 1998]

In the specific case of the MRI scanner we looked into limitations to evolvability and causes. We observed:

- 25 years of historical growth
- Inherent complexity of system and context
- Size and complexity of organization
- Size and complexity of realization
- Lack of overview (large amount of detailed documentation is available)
- Human and cultural factors (high level of expertise, conservatism)

In summary, it is inherently a complex system in a complex environment. Additional complexity grows over time, caused by human, organizational and cultural factors.

Besides the standard challenges of architecting, such as decomposition, interface management, and decoupling, we also observed insufficient underpinning of decisions by value and cost, unbalance in core/key/base technology choices, and diversity of configurations.

Figure 2 shows the annotated project goal, based on the evolvability analysis.

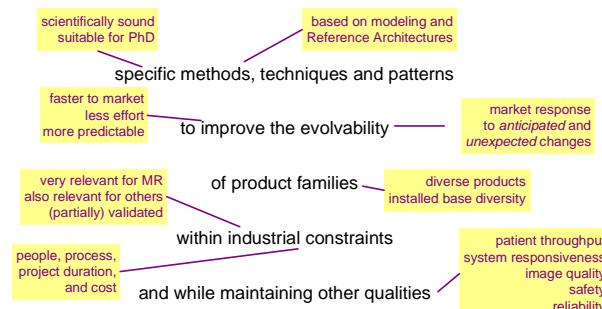


Figure 2 Project Goal

Changes may come from many different sources. Figure 3 shows several examples of changes for MRI scanners. Changes in base technology, such as Windows and PCI-X impact the system architecture and realization. Domain specific changes such as new RF coils or gradient amplifiers also impact the system. Examples of changes in the business are for example the internal organization of the supply of components or a new business

model for viewing. The customer context may induce changes by integration with other systems such as Picture Archiving and Communication (PACS) and Radiology Information (RIS) systems. Legislation changes, such as reimbursement, also impact the system. Finally, applications are a main source of changes.

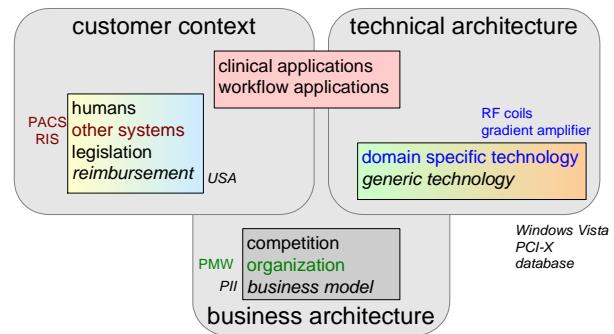


Figure 3 sources of change with some examples

Research Questions

We have transformed the evolvability analysis in a set of research questions as shown in Figure 4. Individual researchers address few of these questions by looking at a part of the total system architecture.

How to transform into an evolvable product family architecture?
How to support decision making? business wise technological
How to create overview? by visualization by high-level modeling
How to mine the realization for implicit know how?
What are practical guidelines? for decomposition for interface definition
What are patterns that support evolvability?

Figure 4 research questions

Why Reference Architectures?

Reference architectures [Muller 2007] extract essential know how from existing architectures, for example by using patterns. The goal of reference architectures is to support decision making, to provide overview and to provide practical guidelines. All five subquestions of the research questions relate directly to reference architectures. We can reformulate the questions into a proposition:

Creation of a reference architecture based on mining existing realizations for know how and patterns will create overview, support decision making, and will provide practical guidelines for design.

Major research challenge is to discover how a reference architecture should look like, since the term is used often, but effective instantiations are scarce. Major challenges are the heterogeneity of information (many involved disciplines), and the degree of abstraction and amount of details (overview is quickly lost when too much detail is present; abstract high level descriptions might not provide any guidance).

Example research project result

The research project itself is staffed with 10 PhD students from different disciplines and partners, 3 research fellows, 1 project leader and 1 knowledge manager, and allocated support from MR architects and engineers, and academic supervisors. The project members work in small teams on related subjects. We will discuss the result of one such team to show the relation between evolvability and Reference Architectures.

One pair of PhD students works on the evolution of communication technology internal in the system. The communication needs in the system increase for many reasons, where an increase in the number of RF receive channels is the most visible. The communication and the digital processing technologies have been changing quite fast the

last decade. This creates many opportunities to change function allocation and to select COTS technologies.

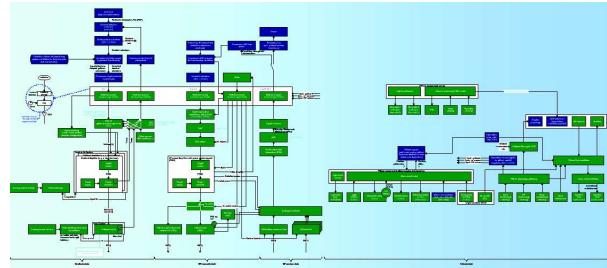


Figure 5, functional architecture diagram

The proposition of this research team is that at least three views need to be combined to tackle this type of communication question:

- Functional diagram
- Physical diagram
- Quantification of the problem

Typically a Reference Architecture will contain many of these diagrams plus many other diagrams. Effectively the two PhD students have created a small part of a Reference Architecture.

Figure 5 shows the functional architecture diagram of the current system. This diagram and the physical architecture diagram have been created by reading documentation and by interviewing experts.

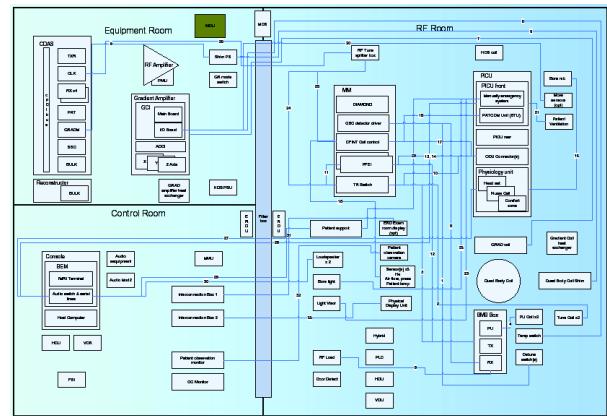


Figure 6, physical architecture diagram

Figure 6 shows the physical architecture diagram. This physical diagram covers three rooms: the technical room, the operator room and the room with the patient and the magnet itself.

These two diagrams, once created by the PhD students, have been used to discuss the quantification with the MR architects. The diagrams greatly facilitated the discussion, because they provide a shared overview. Most of the communication challenges become visible in the physical diagram, in terms of the number of cables and connectors. However, the quantification discussion in terms of needs is more easily done on the basis of the functional diagram.

During this discussion many additional diagrams pop-up that typically are part of a reference architecture, such as typical pulse sequences, coordinate systems, and the relation between phase of RF signals and image quality.

Conclusion

We have shown the challenge to evolve a family of MRI scanner products over time, where changes are triggered by many different stakeholders. Growing size and complexity over a long period causes lower development efficiency and less innovation. We postulate that system architectures that support change for exploration will also support later life cycle changes, such as scaling to volume production and installed base management.

We formulated a set of research questions. We relate these research questions to Reference Architectures and formulate a proposition that Reference Architectures facilitate architecting of evolvable product families. We have shown one example of an evolvable question where part of a reference architecture was created and used beneficially.

Acknowledgements

The contribution of all project members enabled this article. The discussions within the core project team (Pierre America, Pierre van de Laar, Teade Punter, Joland Rutgers, Dave Watts and I) shaped the project vision and approach. The specific example with

functional and physical architectures is the work of Daniel Borches and Alexander Douglas.

References

- E. Fricke and A. Schulz, “Design for Changeability (Dfc): Principles to Enable Changes in Systems Throughout Their Entire Lifecycle”, *Systems Engineering*, Vol. 8, No. 4, 2005, pp. 342-359.
- Pierre van de Laar et al, “The Darwin Project: Evolvability of Software-Intensive System”, ICSM 2007, Paris, October 2007.
- Gerrit Muller and Eirik Hole (editors). “Reference architectures; why, what and how.” White Paper Resulting from Architecture Forum Meeting March 12-13, 2007 (Hoboken NJ, USA). http://www.architectingforum.org/whitepapers/SAF_WhitePaper_2007_4.pdf.
- David L. Parnas, “On the Criteria to Be Used in Decomposing Systems into Modules”, *Communications of the ACM*, 15, 12 December 1972, pp1053-1058
- David L. Parnas, “Some Software Engineering Principles”, Infotech State of the Art Report on Structures Analysis and Design, Infotech International, 1978.
- Colin Potts. “Software-engineering research revisited.” *IEEE Software*, Vol. 10, No. 5:19–28, September/October 1993.
- Rob van Ommering, “Beyond Product Families: Building a Product Population”, IW-SAPF-3, Las Palmas, March 2000
- Jack Ring, Ernst Fricke “Rapid evolution of all your systems- problem or opportunity”, IEEE, 1998.
- Armin P. Schulz, Ernst Fricke, Eduard Igenbergs. “Enabling Changes in Systems throughout the Entire Life-Cycle – Key to Success?”. *Proceedings of the 10th annual INCOSE conference*, July 2000, Minneapolis, USA.

Gerrit Muller received his Master's degree in Physics from the University of Amsterdam in 1979. He worked from 1980 until 1997 at Philips Medical Systems as system architect. From 1997 to 1999 he was manager System Engineering at ASML. From 1999 - 2002 he worked at Philips Research. Since 2003 he is working as senior research fellow at ESI (Embedded Systems Institute). In June 2004 he received his doctorate. The main focus of his work at ESI is on System Architecture methods and on education of future System Architects. January 2008 Gerrit Muller starts as full professor Systems Engineering at Buskerud University College, in Norway. Special areas of interest are:

- Ways to cope with the exponential growth of size and complexity of systems. Examples of methods to address the growing complexity are product lines and composable architectures.
- The human aspects of systems architecting (which in itself is a crucial factor in coping with the above mentioned growth).

More information (System Architecture articles, course material, curriculum vitae) can be found at: Gaudí systems architecting
<http://www.gaudisite.nl/>

