

Reducing the Cost of Requirements Engineering

Colin Hood Systems Engineering
Munich Office
Dorfstr. 12

85253 Erdweg
Germany

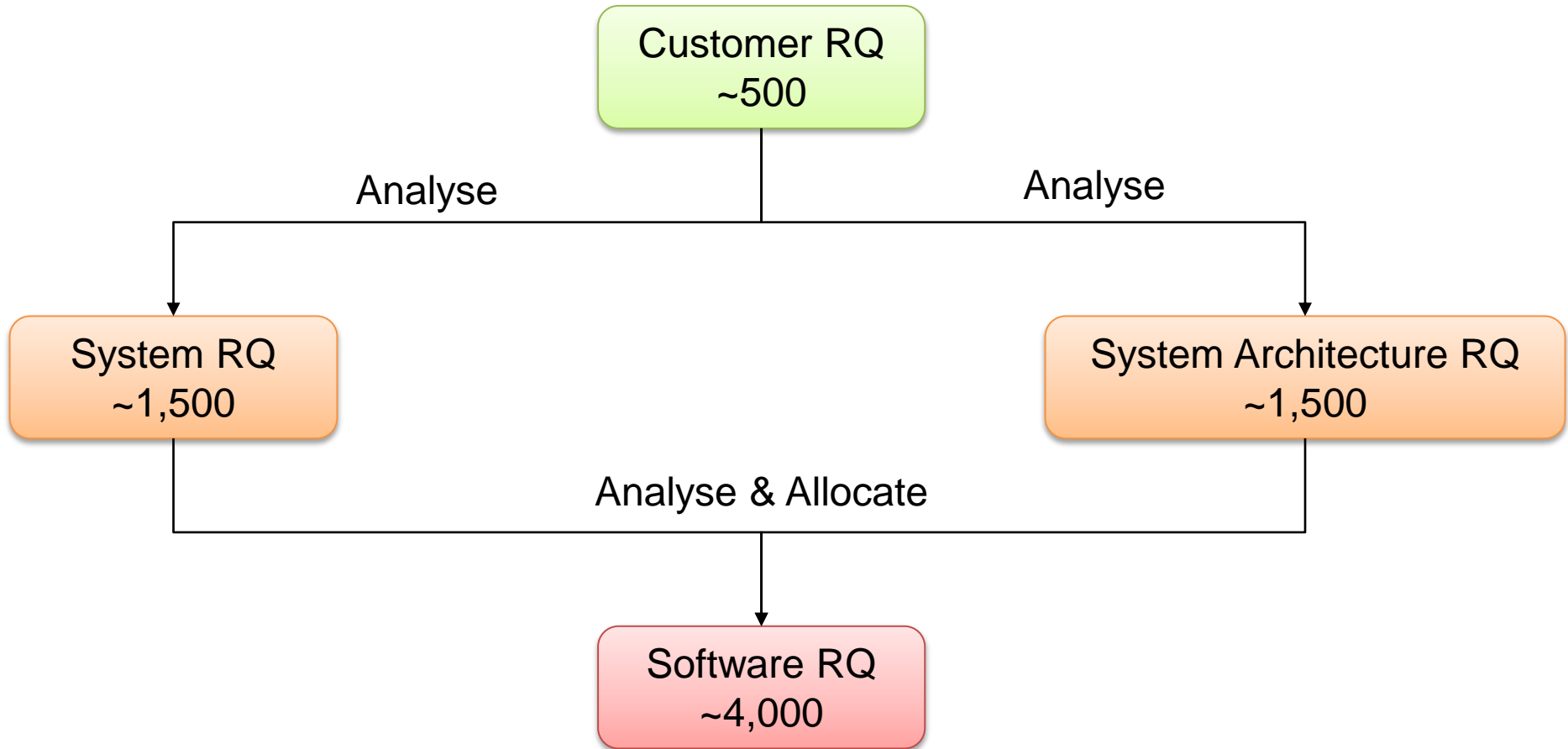
Tel: +49 8138 66 98 625
info@colinhood-se.com

Marco Di Maio
Professor of Systems Engineering
TH Ingolstadt

85049 Ingolstadt
Germany

Tel: +44 795 708 2205
marco.dimaio@thi.de

Case Study: Mid-sized embedded electronics project



7,500 Requirements

Requirements

- Raison d'être:
 - a) to show customer that you understood their needs
 - b) to give the engineers guidance for their development activities
 - c) to provide a basis for verification
- Scope/Types:
 - a) Customer Requirements and Constraints
 - b) System/Architecture Requirements and Constraints
 - c) Domain Requirements and Constraints } Total ~ 7,500 Requirements
- Initial Cost:
 - a) Analysis (understanding) of Customer Requirements
 - b) Creation of Requirement and associated artefacts } 2 Man-Days per Requirement *
- Additional Cost:
 - a) Changes to Requirements
 - b) Communication, negotiation and common understanding
 - c) Retirement of Requirements

* Measured from several industry projects

→ 15,000 Man-Days

→ 75 Man-Years for requirements administration!

Reducing the Cost of Requirements Engineering (RE)

15,000 Man-Days for mid-size project \cong 75 Man-Years

Options

I. Reducing the Number of Requirements:

- a) Reference norms and standards to replace explicit Requirements
- b) Ensure that every Customer Requirement is valid and understood before allowing it
- c) Remove duplicates and reuse Requirement across different areas of responsibility
- d) Disallow breaking down higher level Requirements without
 - domain knowledge
 - responsibility for the cost
- e) Remove incentives to create Requirements in the first place

II. Reducing the Effort per Requirement:

- a) Model Requirements as objects and create all documentation as views on the model
- b) Give Requirements a lifecycle that is independent from their containing documents
- c) Make sure Requirements are correctly related to other elements of MBSE model
- d) Dynamically create System Requirements from a Semantic System Model

I. Reducing the Number of Requirements (Quick Wins)

1) By concentrating on:

a) **Purpose:** What is to be achieved?

b) **Interfaces:** Signal, information, electrical, or energy (physical impact, e.g. heat or radar).

2) By minimising or avoiding them altogether:

a) **Constraints:** Remove as many constraints as possible to allow for more variants. Make sure you investigate the reason for their inclusion (e.g. required by customer or law) and to eradicate all that are not justified. Let the designers do their job.

b) **Performance Requirement:** These are difficult to specify and should only be defined, if the authors

- are qualified to do so, e.g. the authors can calculate required performance.
- do carry the responsibility for the ensuing costs

3) By referencing standards (rather than copying their contents)

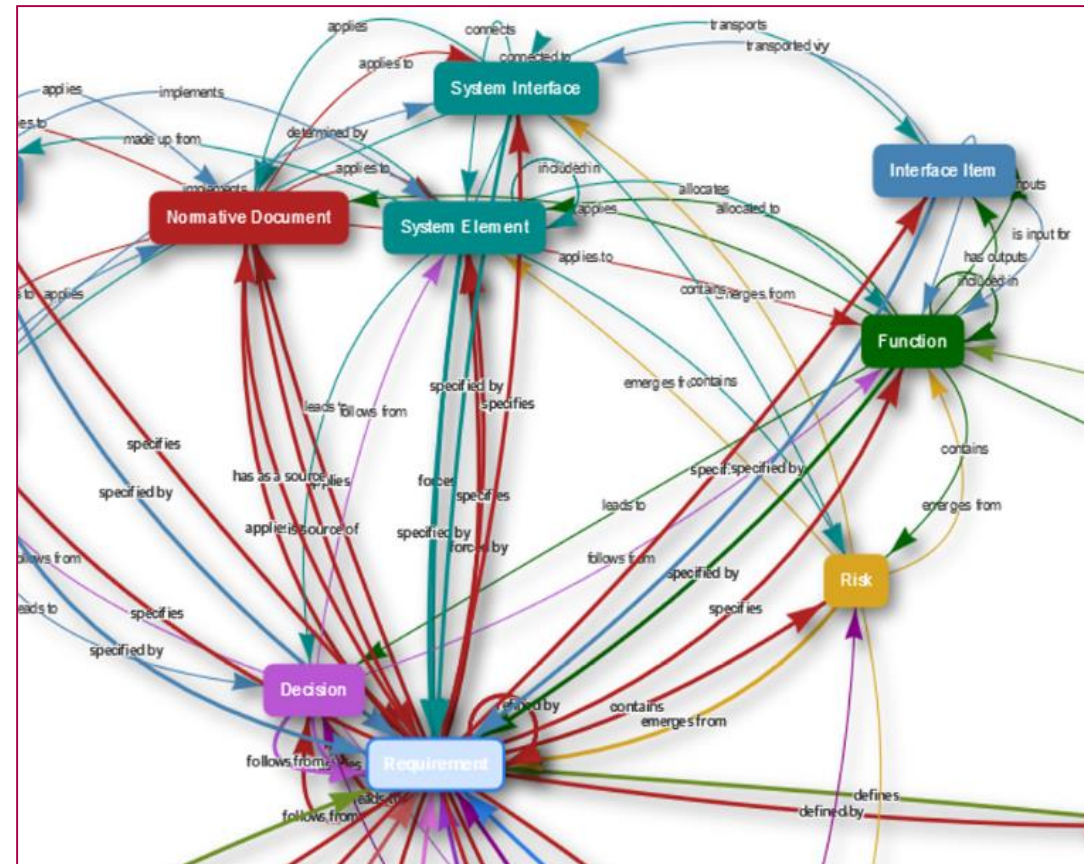
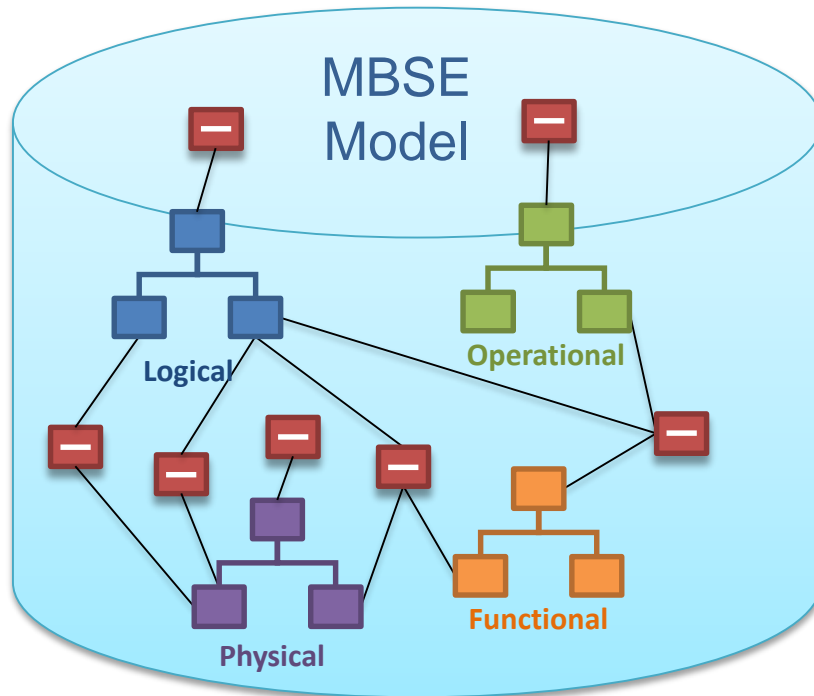
3) By removing duplications: System requirements, system architecture requirements, and discipline requirements are often identical; System requirements are created from the system architecture and often identical to the requirements allocated to system elements in the architecture. System requirements allocated to a discipline are requirements for this discipline, i.e. a system requirement allocated to SW is a SW requirement.

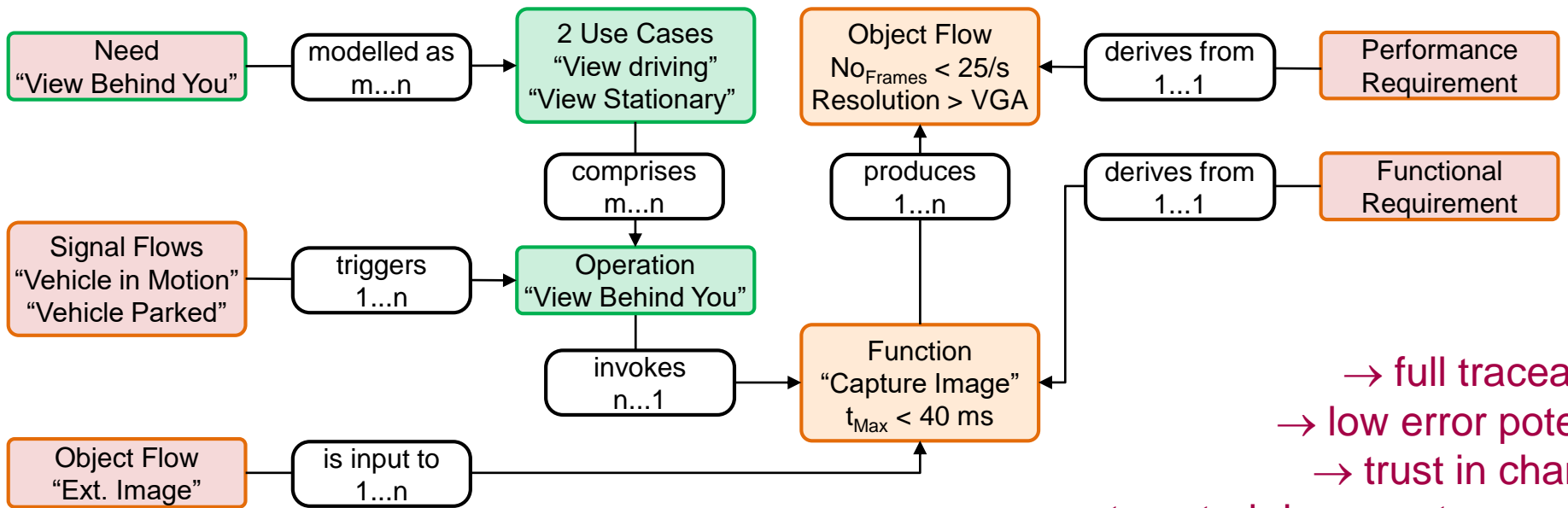
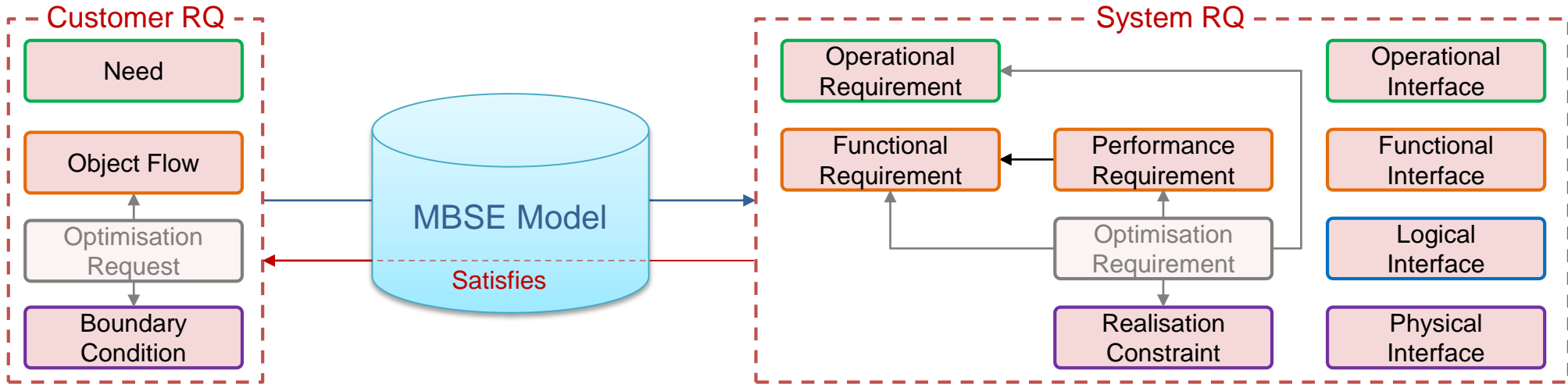
This is not the case for derived or newly created requirements.

II. Reducing the Effort per Requirement

By **Modelling Requirements** as Objects in a semantic MBSE-Model

- consistent model including domains and RQs
- risk reduction (fewer manual processes)
- higher efficiency and efficacy
- automatic traceability across all elements





- full traceability
- low error potential
- trust in changes
- automated document generation

Conclusion

Reducing the Number of Requirements

- In this example project, 4,000 SW Requirements can easily be replaced by 1,500 Requirements that are applicable across multiple levels of the V-model, with no loss of information!
- In total, **5,500 Requirements were useless** rendering **55 Man-Years useless!**

There are exceptions; but from experience this is applicable for **over 90% of Requirements!**
[within embedded electronics projects]

Reducing the Effort per Requirement

- Working with a semantic model provides numerous advantages for MBSE
- Creating Requirements as (small) model objects reduces their initial cost
- Full traceability, logical checks and filtering reduces the additional lifecycle cost (change!)
- Automatic document generation reduces errors, risk and the ensuing costs
- The remaining **1,500 Requirements** could therefore be **derived from the model**