

The Role and Task of the System Architect

by *Gerrit Muller* Buskerud University Collge

e-mail: gaudisite@gmail.com

www.gaudisite.nl

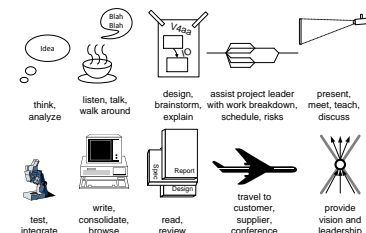
Abstract

The role of the system architect is described from three viewpoints: deliverables, responsibilities and activities. This description shows the inherent tension in this role: a small set of hard deliverables, covering a fuzzy set of responsibilities, hiding an enormous amount of barely visible day-to-day work.

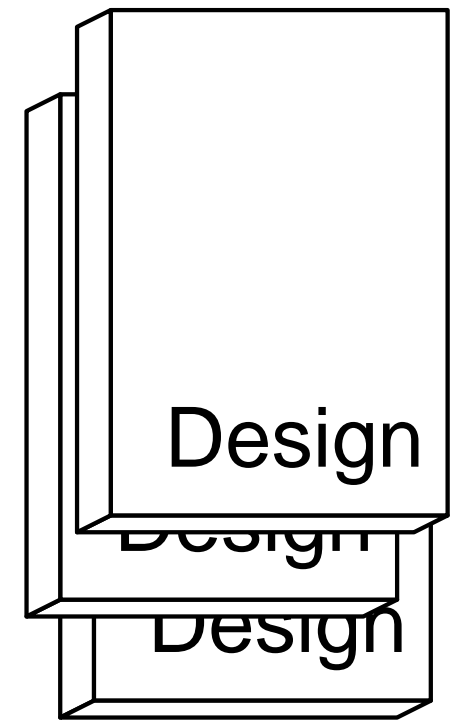
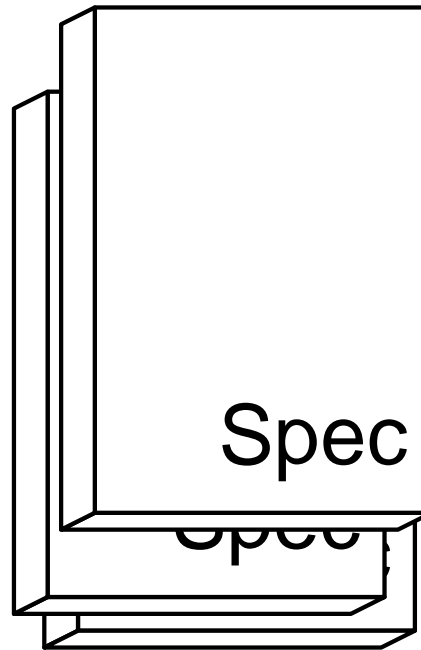
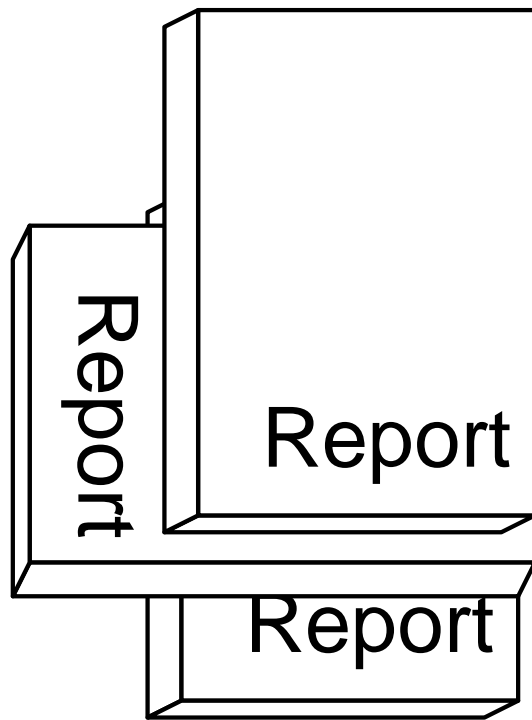
Distribution

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.

October 20, 2017
status: concept
version: 2.0



Deliverables of the System Architect



List of Deliverables

Customer and Life-Cycle Needs (*what is needed*)

System Specification (*what will be realized*)

Design Specification (*how the system will be realized*)

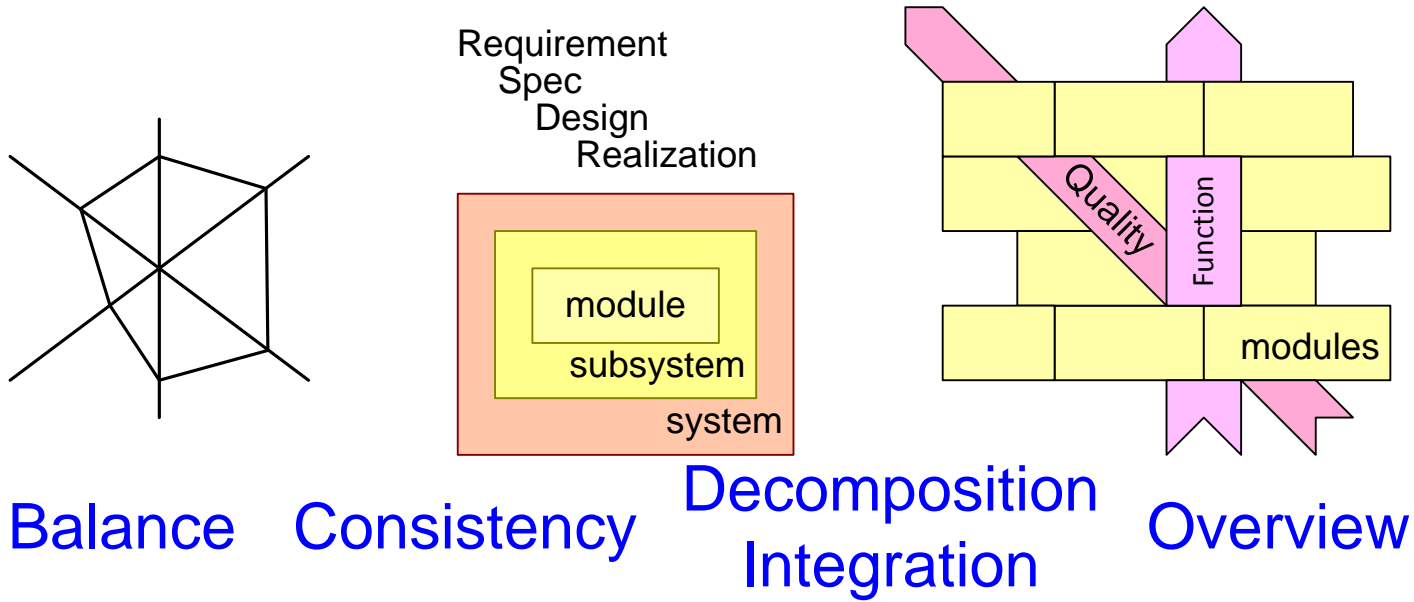
Verification Specification (*how the system will be verified*)

Verification Report (*the result of the verification*)

Feasibility Report (*the results of a feasibility study*)

Roadmap

Responsibilities of the System Architect

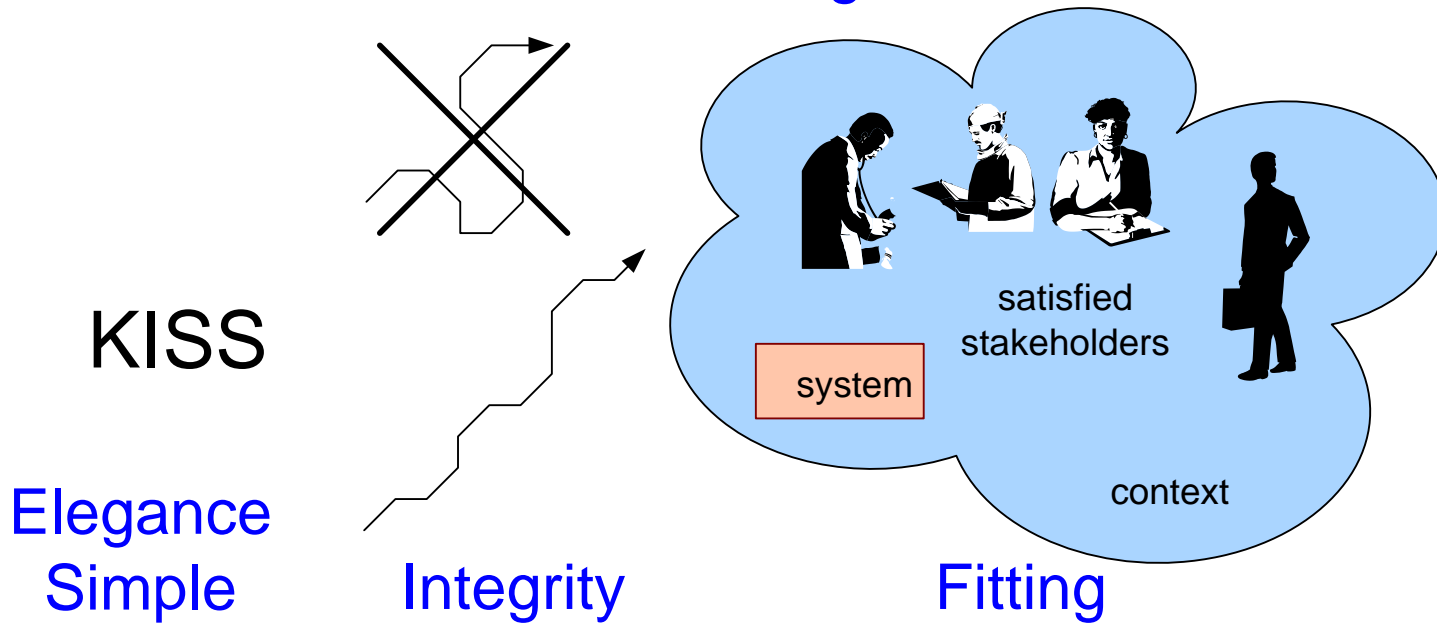


Balance

Consistency

Decomposition
Integration

Overview



KISS

Elegance
Simple

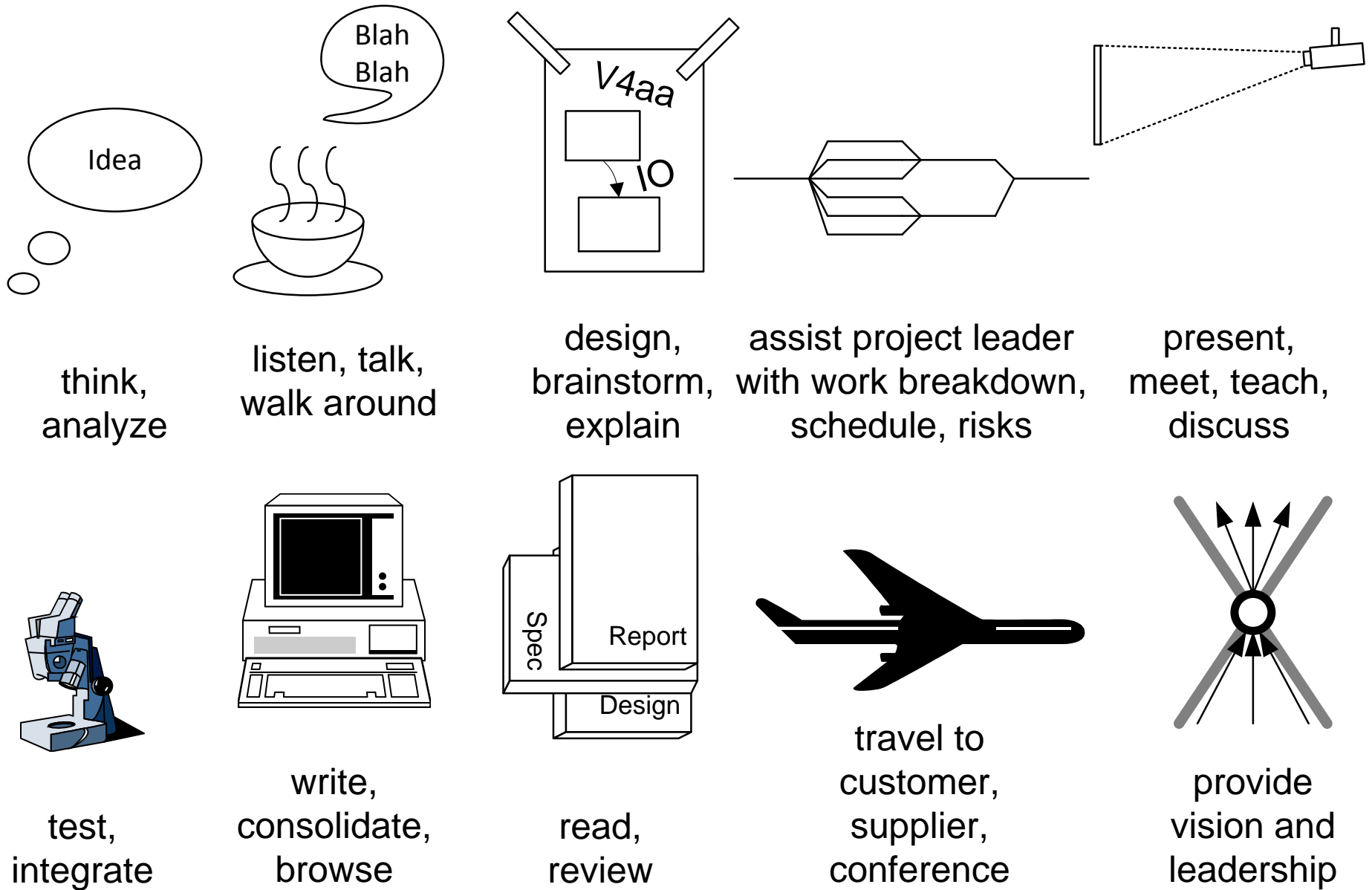
Integrity

Fitting

Examples of Secondary Responsibilities

responsibility	primary owner
business plan, profit	business manager
schedule, resources	project leader
market, saleability	marketing manager
technology	technology manager
process, people	line manager
detailed designs	engineers

What does the System Architect do?



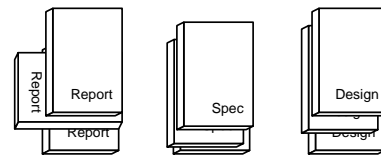
From Detail to Overview

		Quantity per year (order-of- magnitude)	architect time per item
consolidation in deliverables meetings informal contacts sampling scanning	→ driving views	10	100 h
	→ shared issues	10^2	1 h
	→ touched details	10^4	0.5 – 10 min
	→ seen details	$10^5 - 10^6$	0.1 – 1 sec
	→ product details	$10^7 - 10^{10}$	
	real-world facts	infinite	

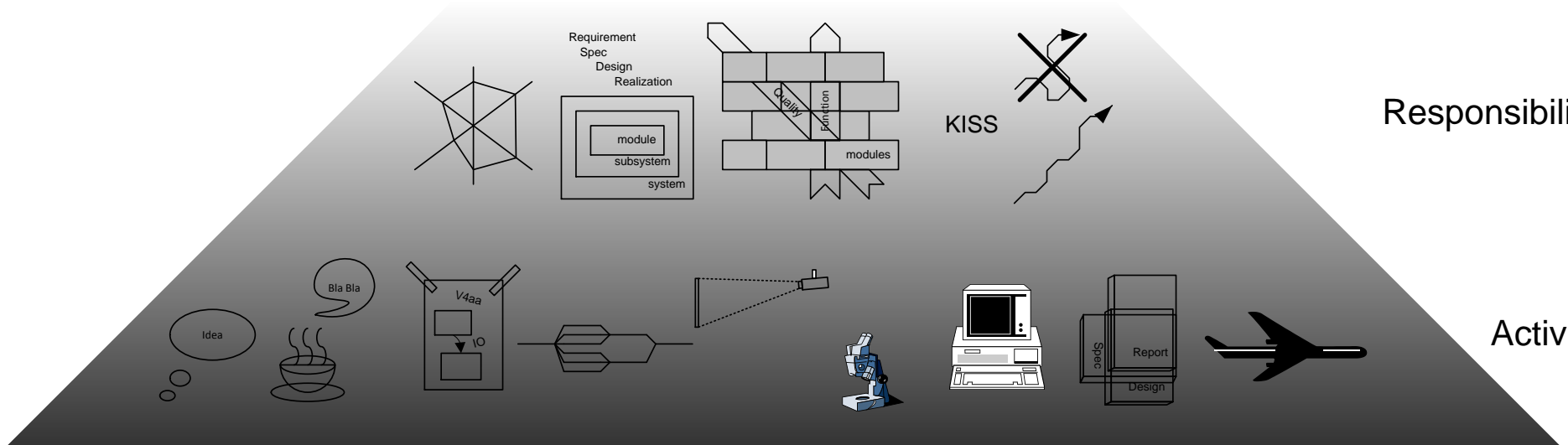
Abstractions only exist for concrete facts.

Visible Output versus Invisible Work

From Manager perspective



Deliverables



Responsibilities

Activities