

Module Functional View

by *Gerrit Muller* University of South-Eastern Norway-NISE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

This module addresses the Functional View.

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.

August 16, 2025

status: draft

version: 0

logo

TBD

The functional view

by *Gerrit Muller* University of South-Eastern Norway-NISE

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

Abstract

The purpose of the functional view is described. A number of methods or models is given to use in this view: (use) case descriptions, commercial decomposition function and feature specifications performance models and specifications, information models. The role of standards is discussed.

Distribution

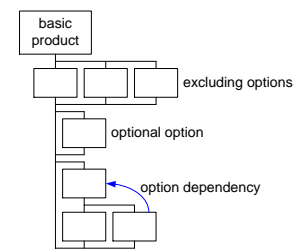
This article or presentation is written as part of the Gaudí project. The Gaudí project philosophy is to improve by obtaining frequent feedback. Frequent feedback is pursued by an open creation process. This document is published as intermediate or nearly mature version to get feedback. Further distribution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

August 16, 2025

status: preliminary

draft

version: 1.0



Example personal video recorder use case contents

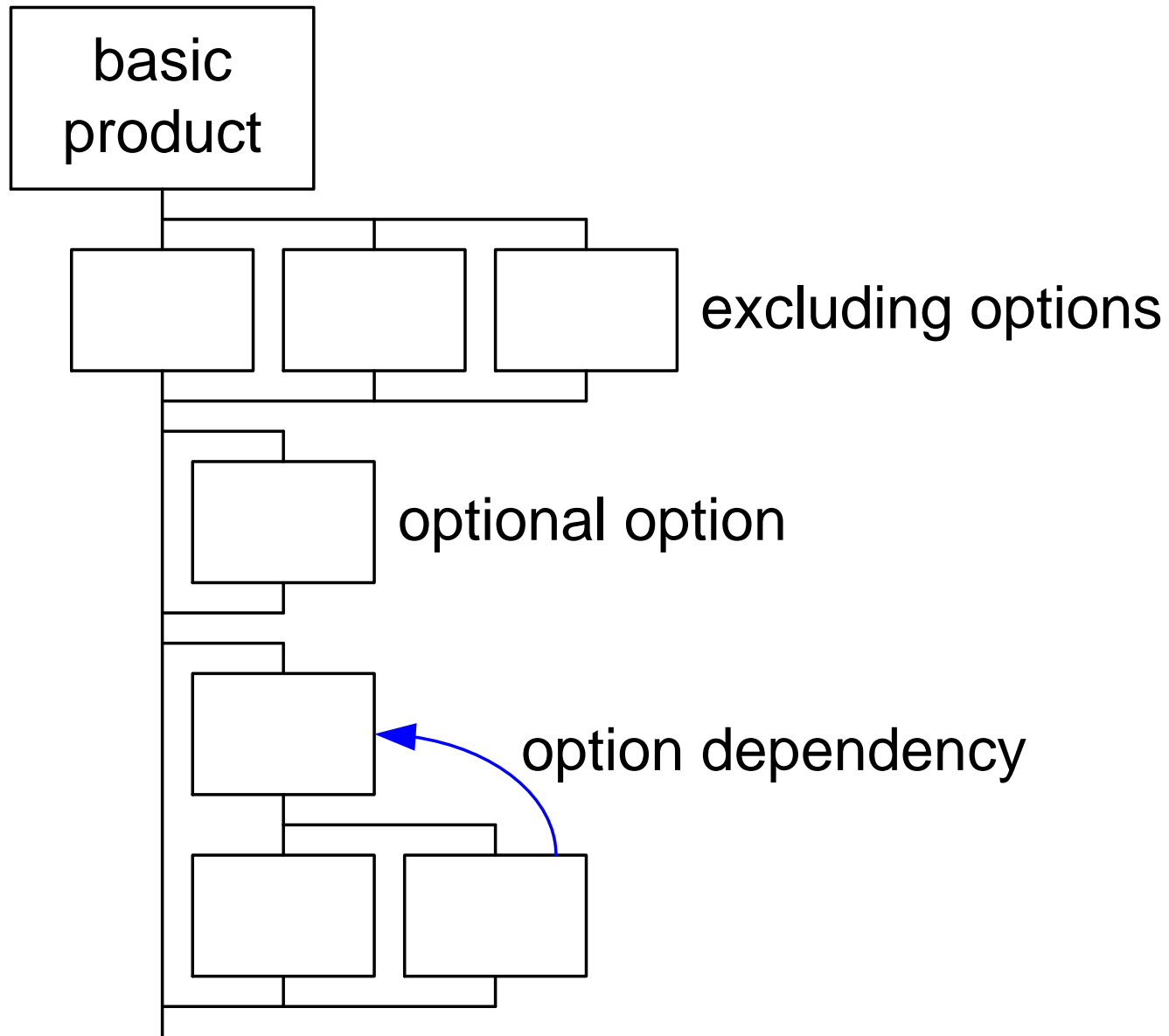
typical use case(s)	worst case, exceptional, or change use case(s)
<p data-bbox="168 496 1012 544">interaction flow (functional aspects)</p> <ul data-bbox="219 555 960 804" style="list-style-type: none"><li data-bbox="219 555 703 596">select movie via directory<li data-bbox="219 608 432 649">start movie<li data-bbox="219 660 680 702">be able to pause or stop<li data-bbox="219 713 898 754">be able to skip forward or backward<li data-bbox="219 766 607 807">set recording quality	<p data-bbox="1131 496 1346 544">functional</p> <ul data-bbox="1182 555 1783 754" style="list-style-type: none"><li data-bbox="1182 555 1783 596">multiple inputs at the same time<li data-bbox="1182 608 1554 649">extreme long movie<li data-bbox="1182 660 1742 702">directory behaviour in case of<li data-bbox="1227 713 1756 754">extreme many short movies
<p data-bbox="168 852 875 963">performance and other qualities (non-functional aspects)</p> <ul data-bbox="219 975 934 1174" style="list-style-type: none"><li data-bbox="219 975 786 1016">response times for start / stop<li data-bbox="219 1027 934 1069">response times for directory browsing<li data-bbox="219 1080 674 1121">end-of-movie behaviour<li data-bbox="219 1133 927 1174">relation recording quality and storage	<p data-bbox="1131 852 1444 900">non-functional</p> <ul data-bbox="1182 911 1973 1158" style="list-style-type: none"><li data-bbox="1182 911 1823 952">response time with multiple inputs<li data-bbox="1182 963 1807 1005">image quality with multiple inputs<li data-bbox="1182 1016 1594 1058">insufficient free space<li data-bbox="1182 1069 1973 1110">response time with many directory entries<li data-bbox="1182 1121 1807 1163">replay quality while HQ recording

Recommendations for working with use cases

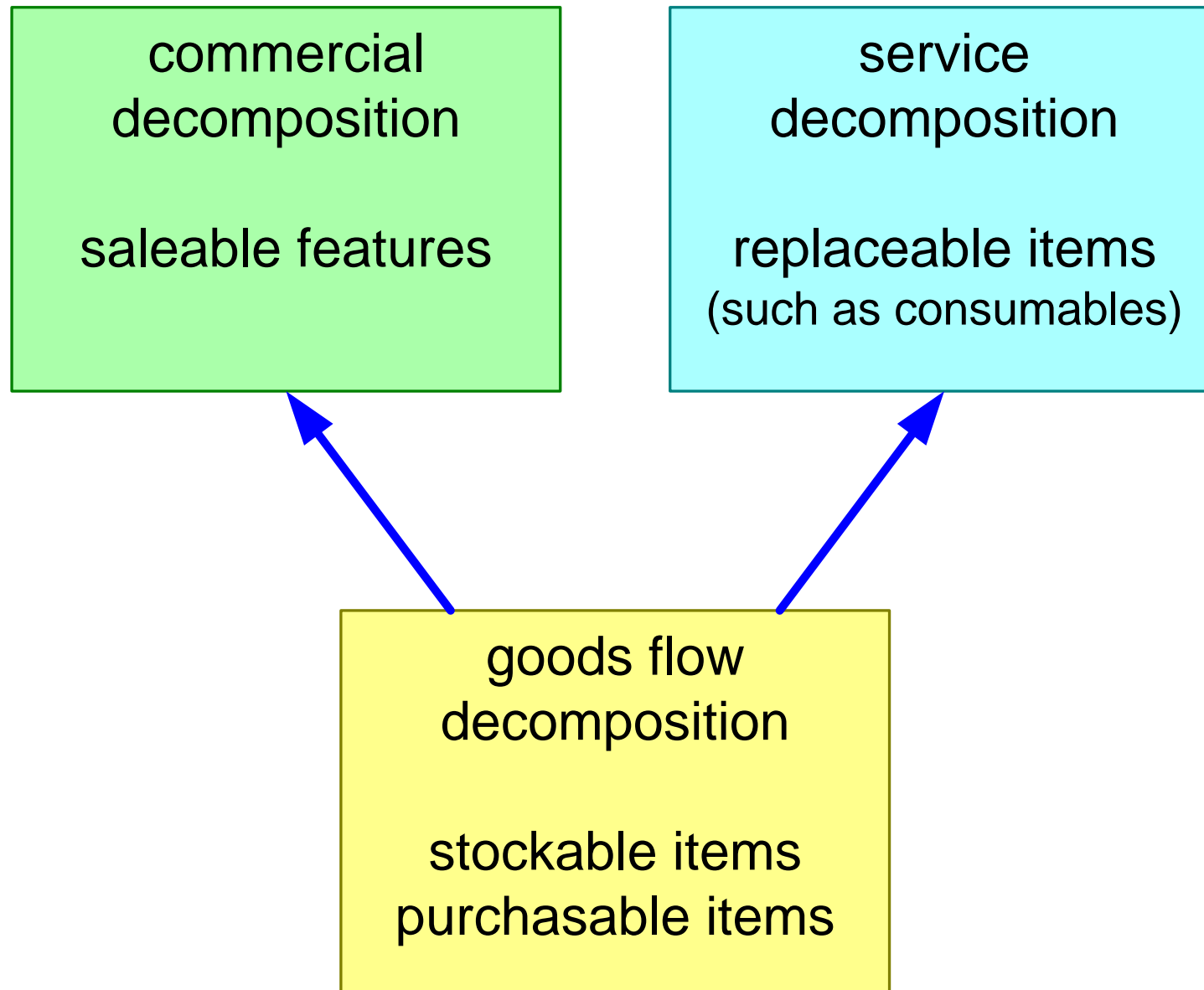
- + combine related functions in one use case
- do not make a separate use case for every function
- + include non-functional requirements in the use cases

- + minimise the amount of required *worst case* and *exceptional use cases*
- excessive amounts of use cases propagate to excessive implementation efforts
- + reduce the amount of these use cases in steps
- a few well chosen *worst case* use cases simplifies the design

Commercial Decomposition



Logistic decompositions for a product

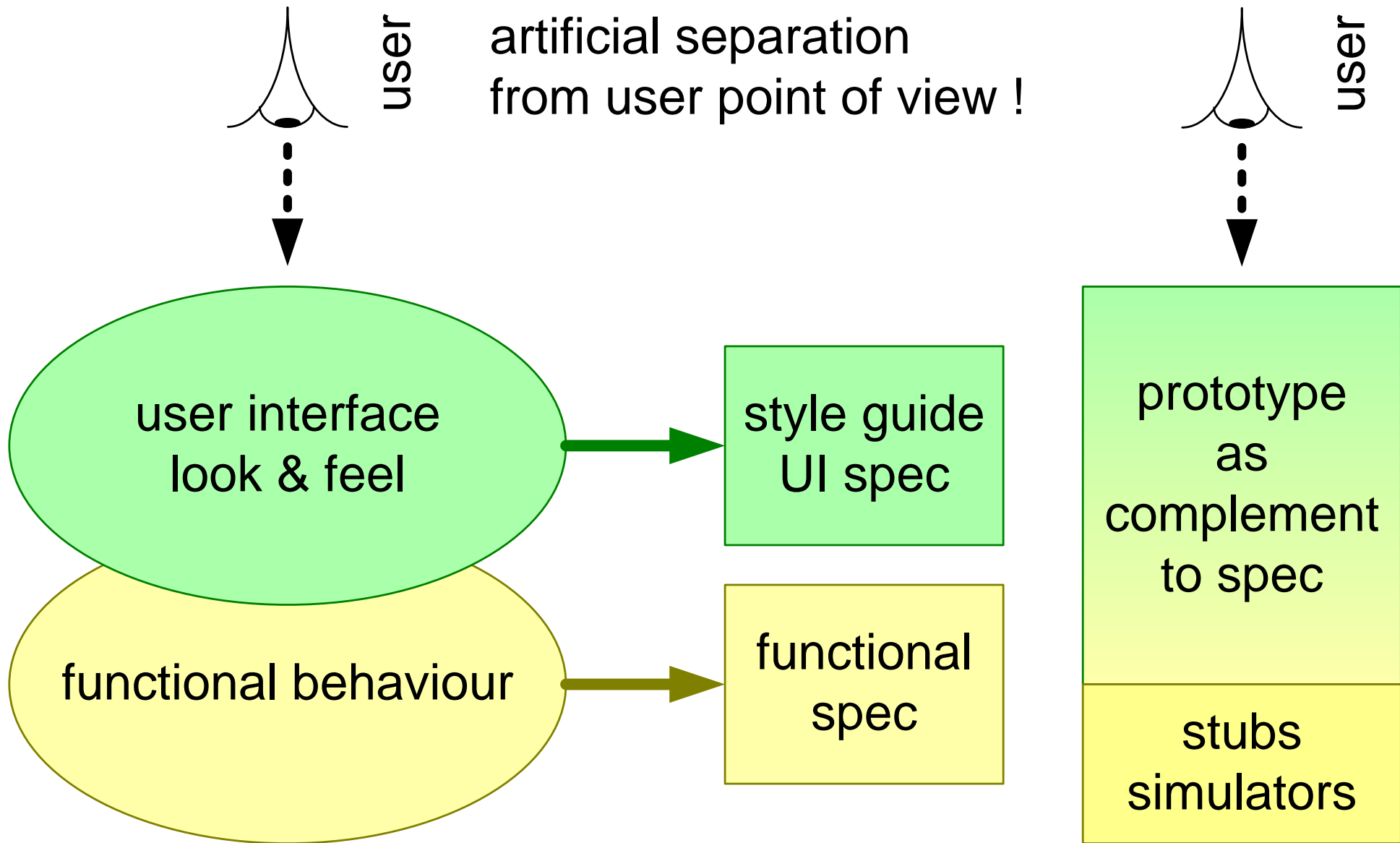


Mapping technical functions on products

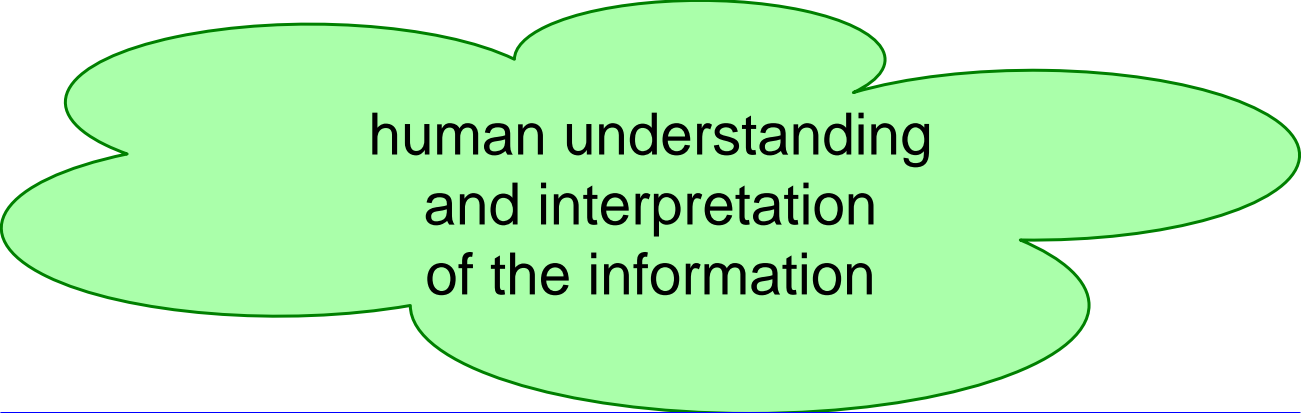
<i>technical functions</i>	<i>products</i>	home cinema system	flat screen cinema TV	bedroom TV
HD display		+	+	-
SD->HD up conversion		+	+	-
HD->SD down conversion		+	+	0
HD storage		0	-	-
SD storage		0	-	0
HD IQ improvement		+	+	-
SD IQ improvement		+	+	+
HD digital input		+	+	0
SD digital input		+	+	0
SD analog input		0	+	+
6 HQ channel audio		+	0	-
2 channel audio		-	+	+

legend	
+	present
0	optional
-	absent

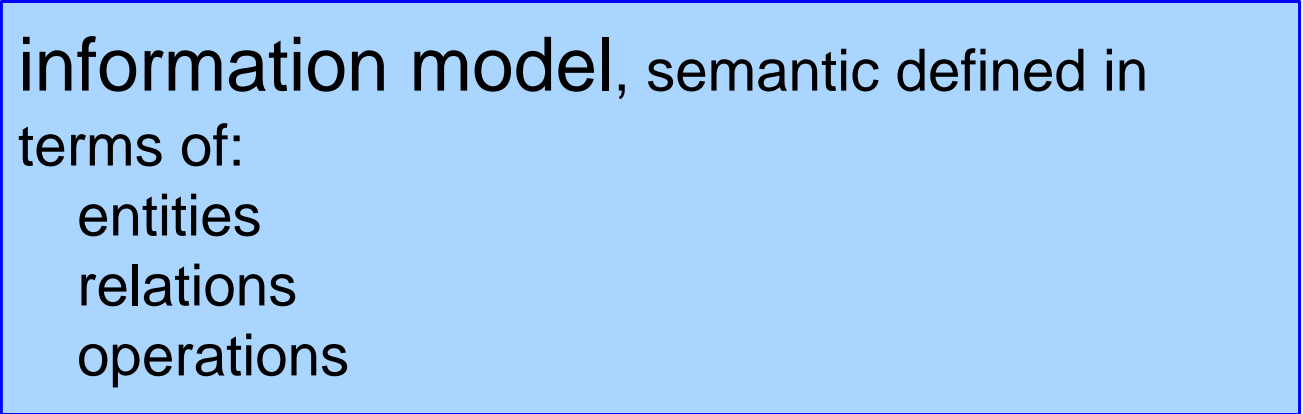
Relation between user interface and functional specification



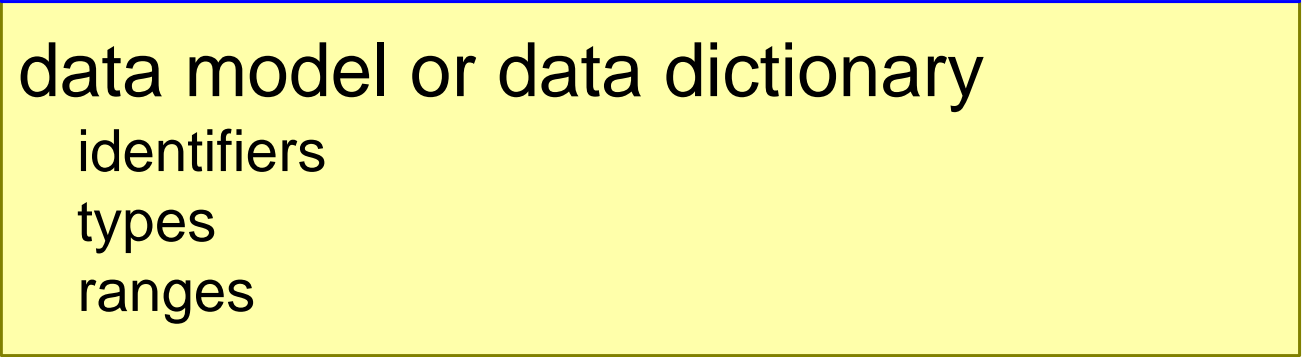
Layering of information definitions



human understanding
and interpretation
of the information

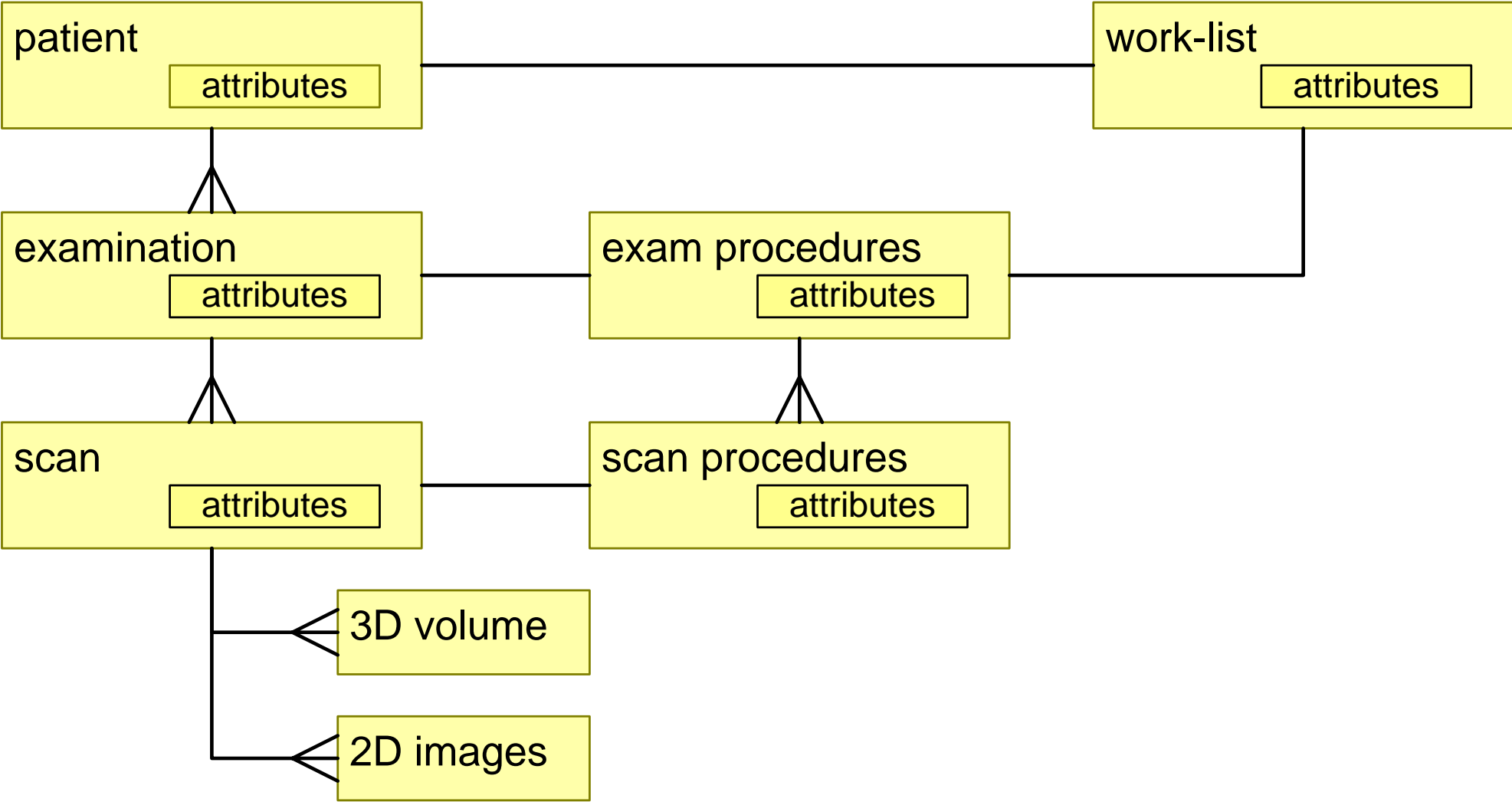


information model, semantic defined in
terms of:
entities
relations
operations



data model or data dictionary
identifiers
types
ranges

Example partial internal information model



12 bit Image:

nx: 16 bit unsigned integer

ny: 16 bit unsigned integer

pixels[nx][ny]: 16 bit unsigned integers [0..4095]

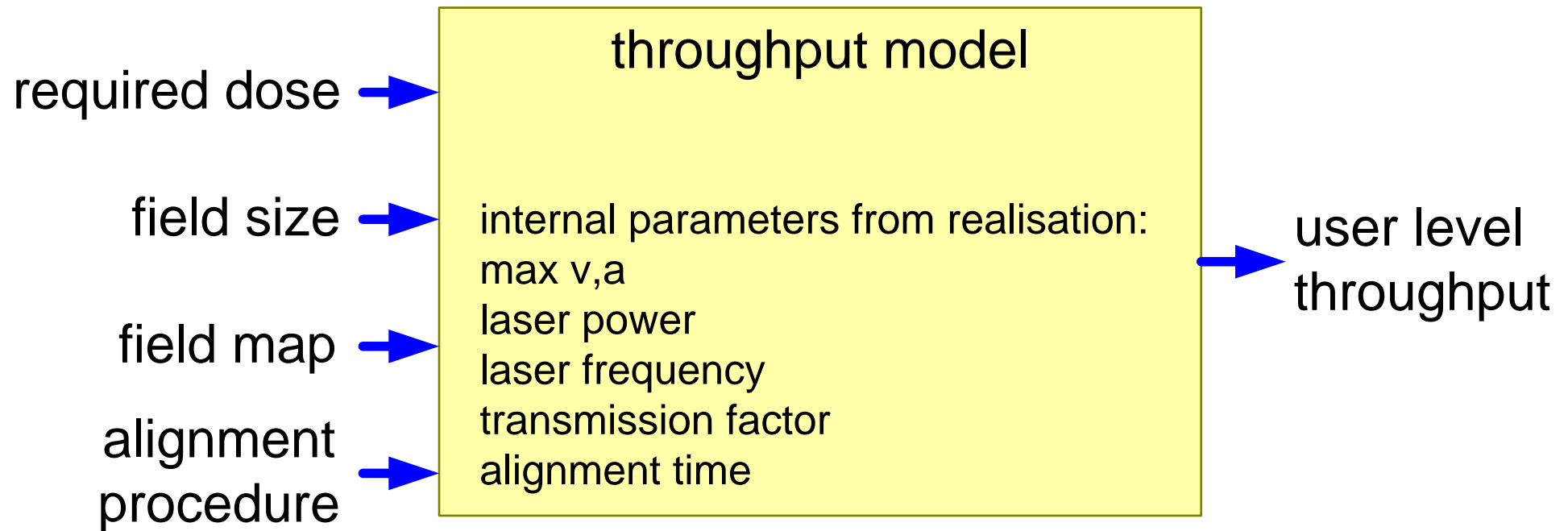
16 bit Image:

nx: 16 bit unsigned integer

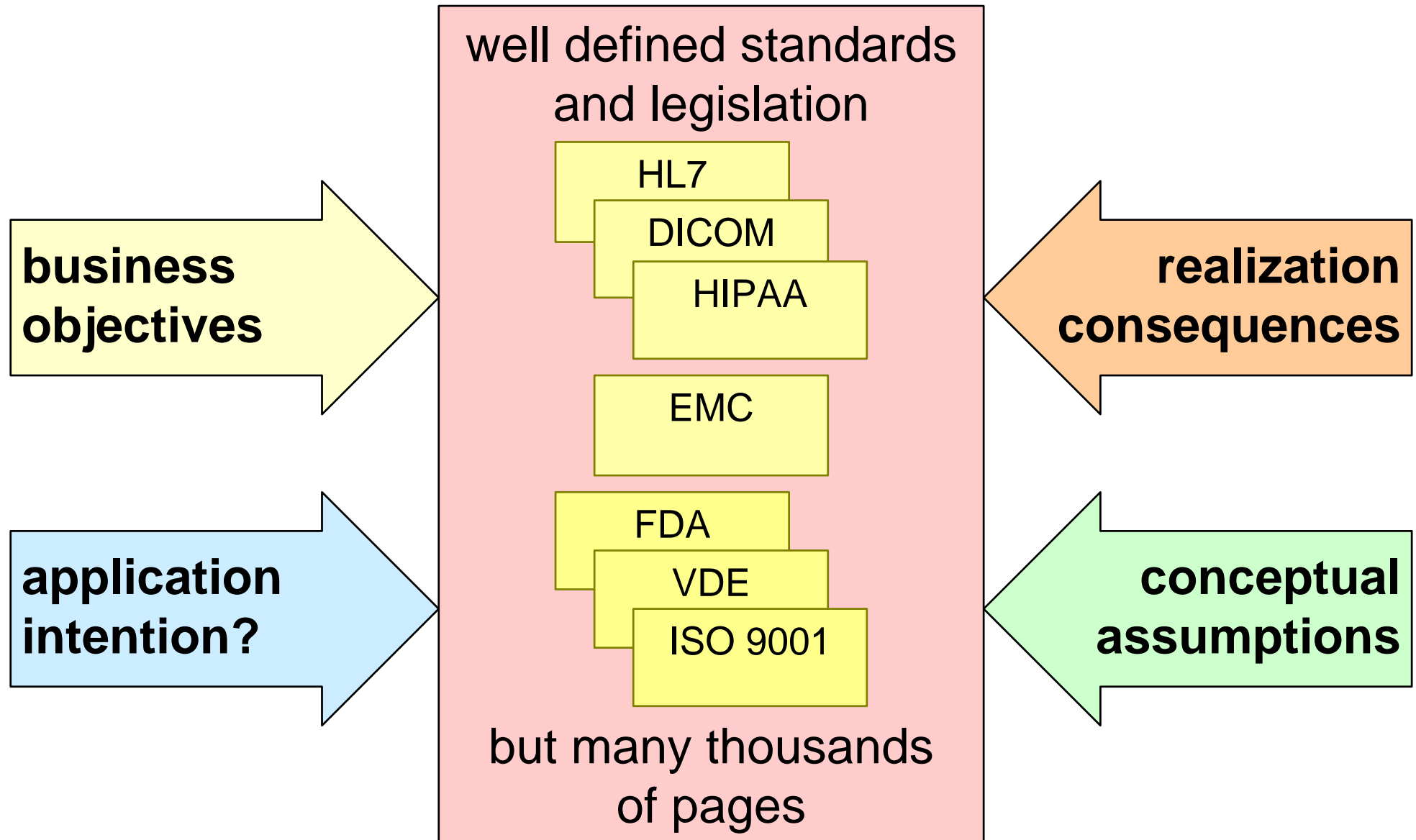
ny: 16 bit unsigned integer

pixels[nx][ny]: 16 bit unsigned integers

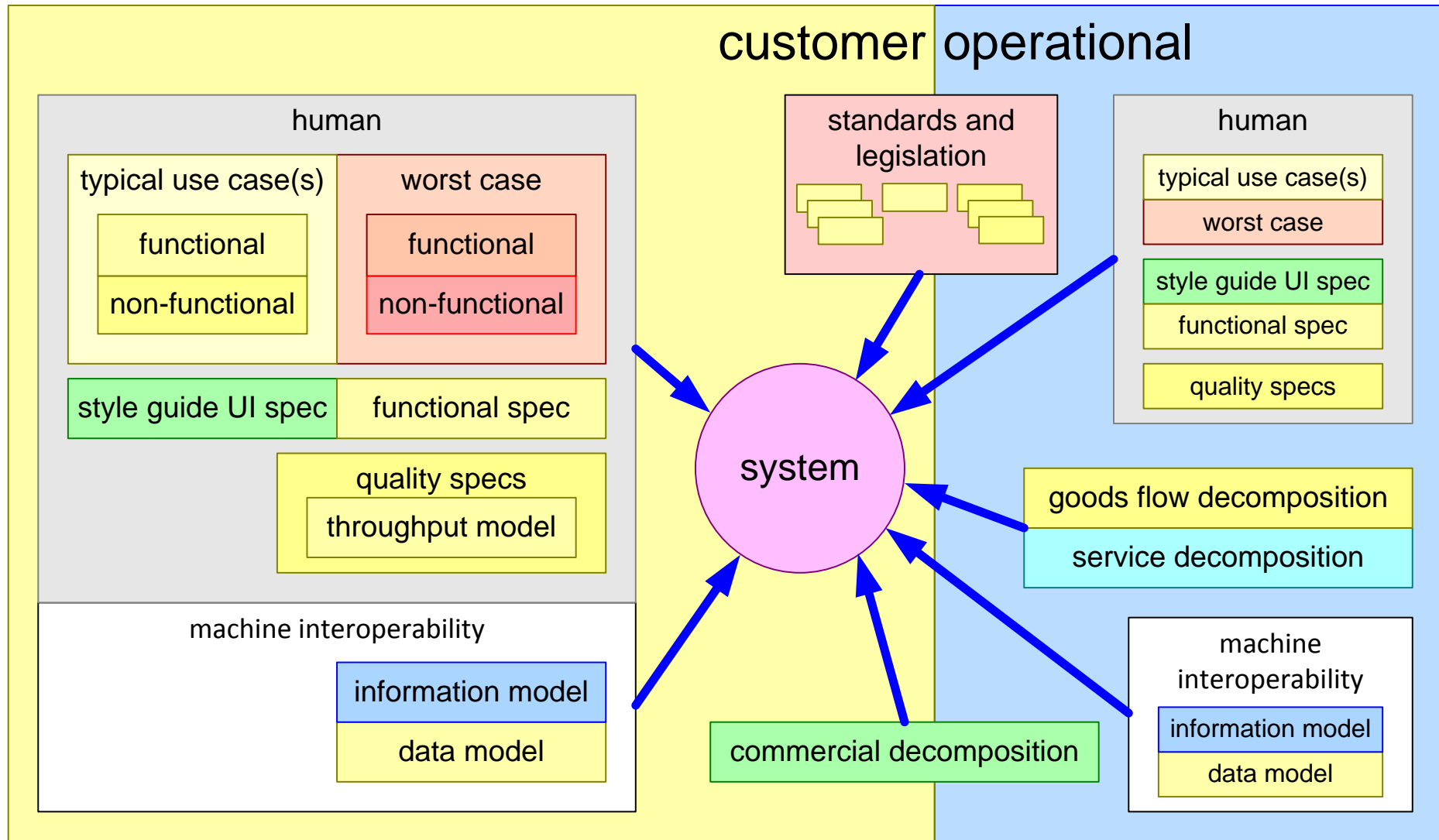
Example of performance modelling



The role of standards



Functional view summary



Functional view = What: externally observable

Exercise Functional View

- Make an overview of functions, performance figures, interfaces and optional features
- identify "most important" (related to CA-views)
- identify "most challenging" (related to CR-views)
- explain why "most important" or "most challenging"
- present in 5 minutes

Goals:

- create awareness of the breadth of the specification
- share the spec with the team
- create a "living" image of the Functional view

Exercise Functional View, second iteration

- Define a typical case, both functions and quantitative
- Create a single page product specification
- Define a worst case, suitable for design exploration and verification