

The functional view

by *Gerrit Muller* Buskerud University College

e-mail: `gaudisite@gmail.com`

`www.gaudisite.nl`

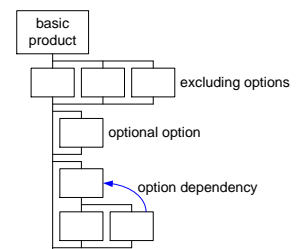
Abstract

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.

October 20, 2017
status: preliminary
draft
version: 1.0



Example personal video recorder use case contents

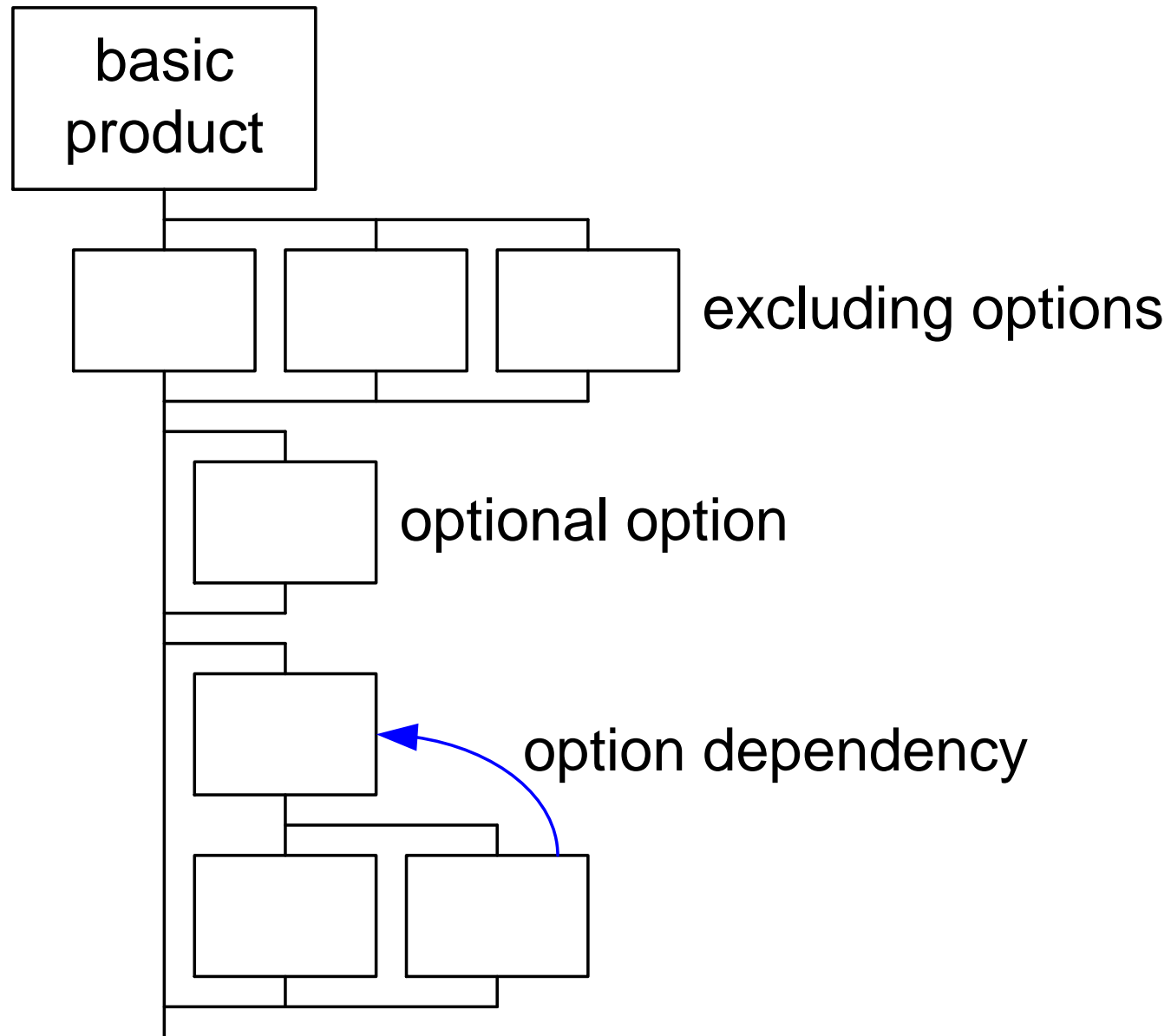
typical use case(s)	worst case, exceptional, or change use case(s)
<p data-bbox="201 505 953 553">interaction flow (functional aspects)</p> <ul data-bbox="247 565 905 805" style="list-style-type: none">select movie via directorystart moviebe able to pause or stopbe able to skip forward or backwardset recording quality	<p data-bbox="1125 505 1331 553">functional</p> <ul data-bbox="1171 565 1755 753" style="list-style-type: none">multiple inputs at the same timeextreme long moviedirectory behaviour in case of extreme many short movies
<p data-bbox="201 849 877 954">performance and other qualities (non-functional aspects)</p> <ul data-bbox="247 966 936 1154" style="list-style-type: none">response times for start / stopresponse times for directory browsingend-of-movie behaviourrelation recording quality and storage	<p data-bbox="1125 849 1430 898">non-functional</p> <ul data-bbox="1171 909 1934 1146" style="list-style-type: none">response time with multiple inputsimage quality with multiple inputsinsufficient free spaceresponse time with many directory entriesreplay quality while HQ recording

Recommendations for working with use cases

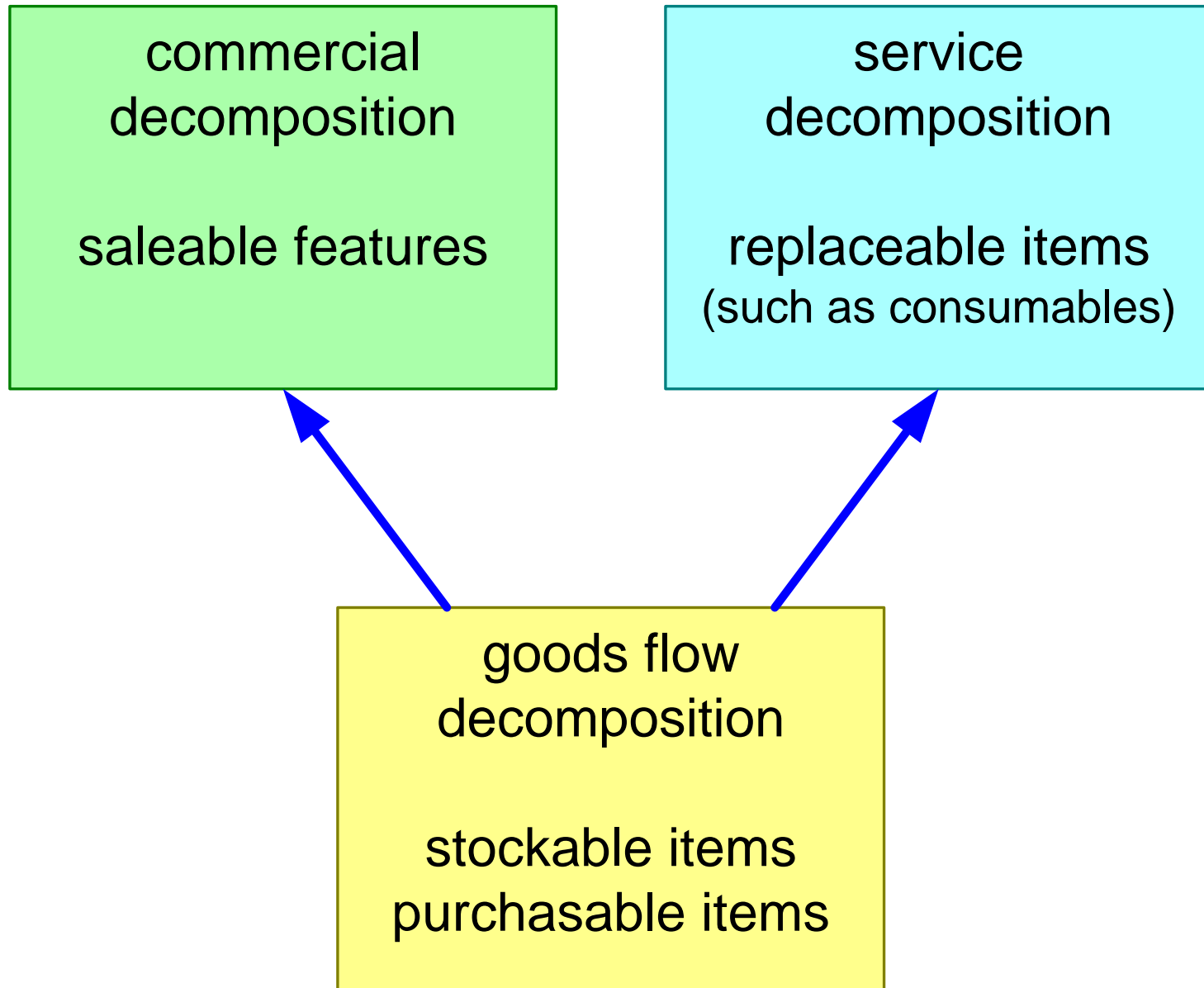
- + combine related functions in one use case
- do not make a separate use case for every function
- + include non-functional requirements in the use cases

- + 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

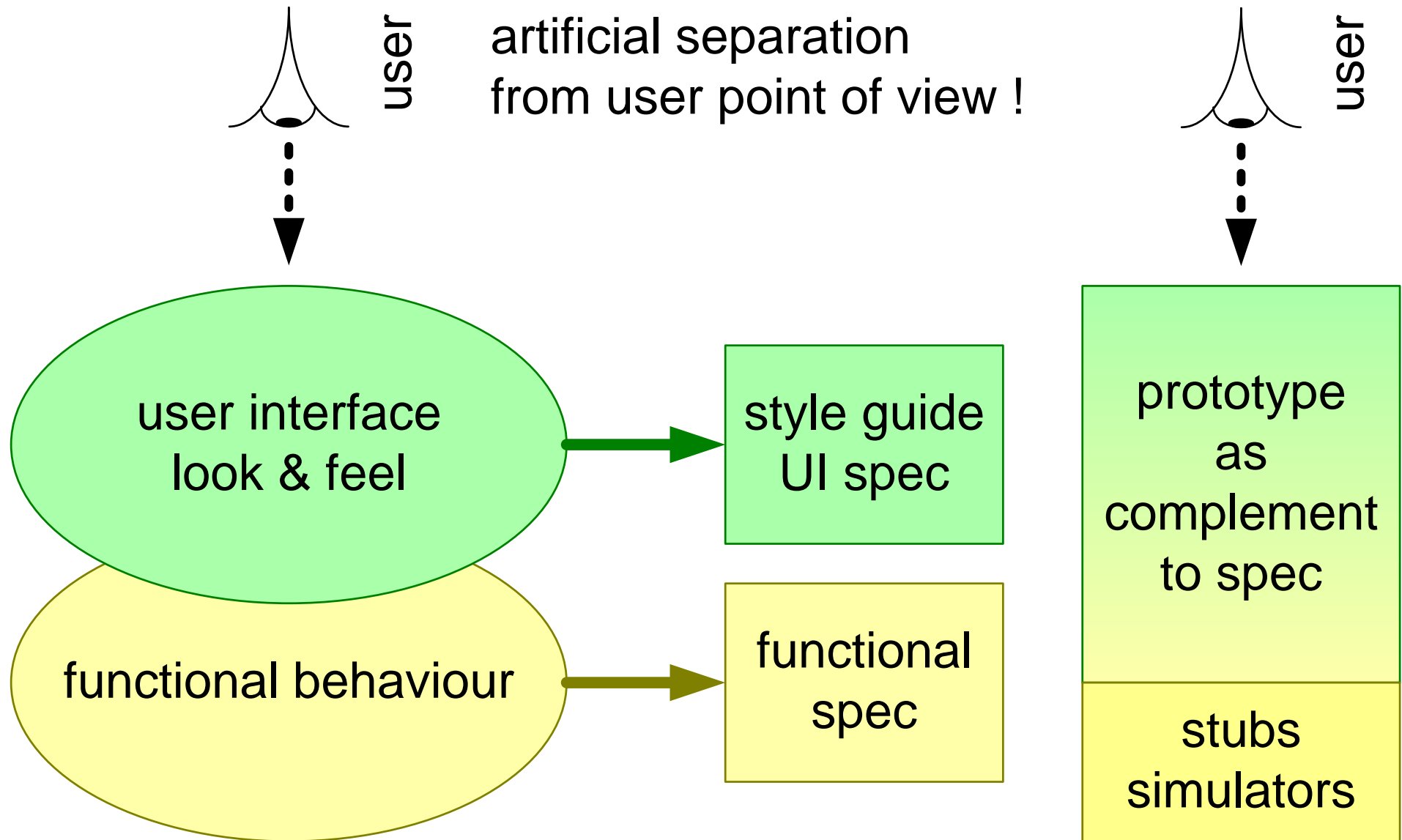
<i>technical functions</i>	<i>products</i>	home cinema system	flat screen cinema TV	bedroom TV
----------------------------	-----------------	--------------------	-----------------------	------------

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



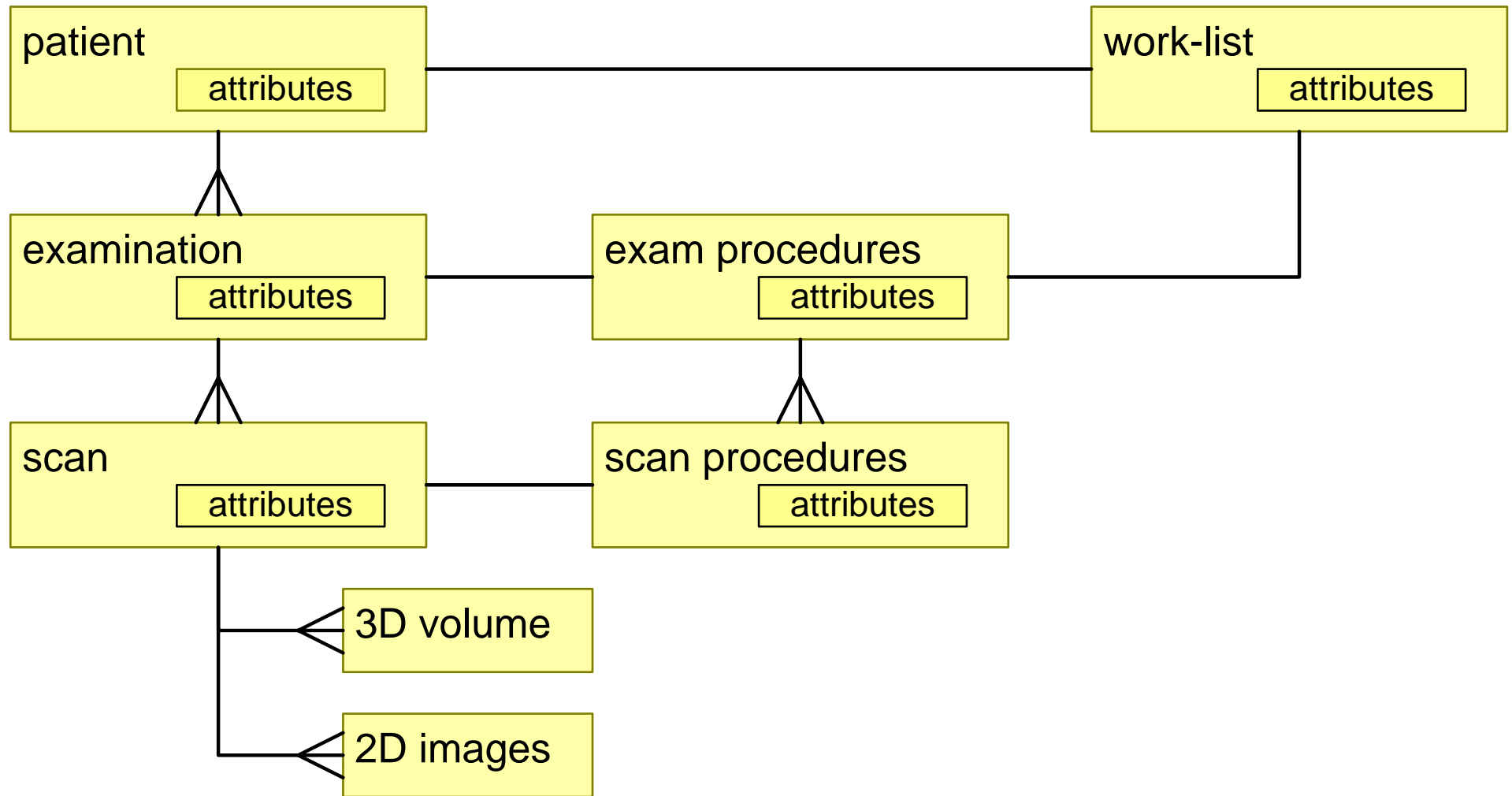
Layering of information definitions

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]

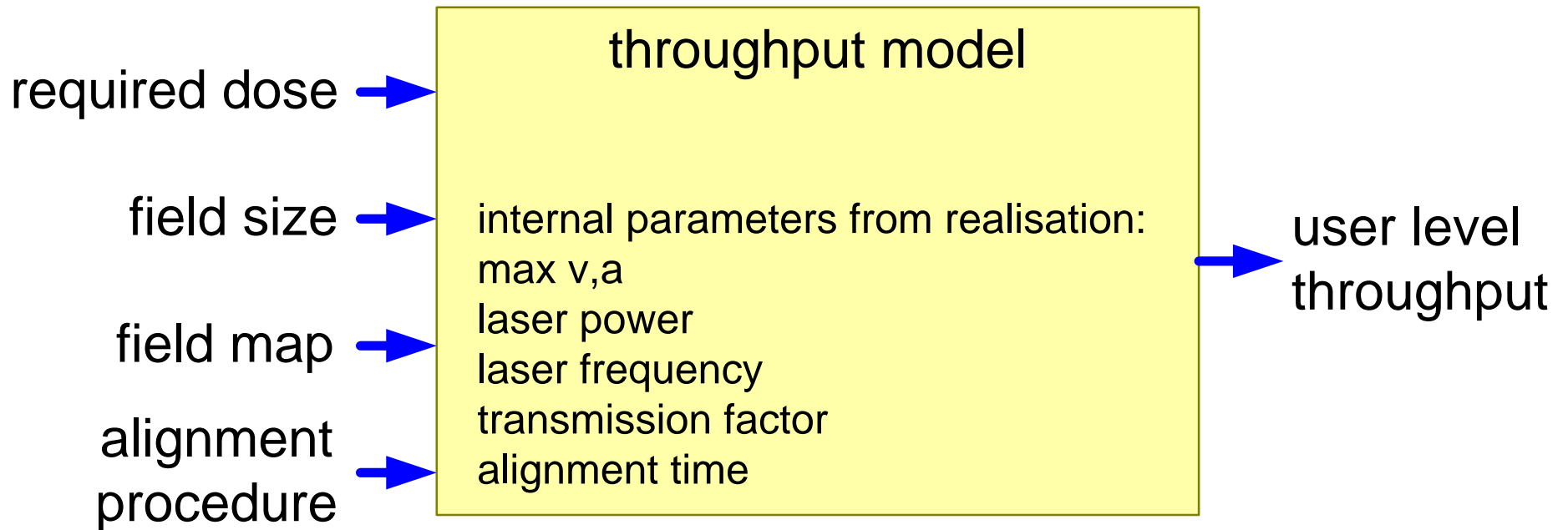
16 bit Image:

nx: 16 bit unsigned integer

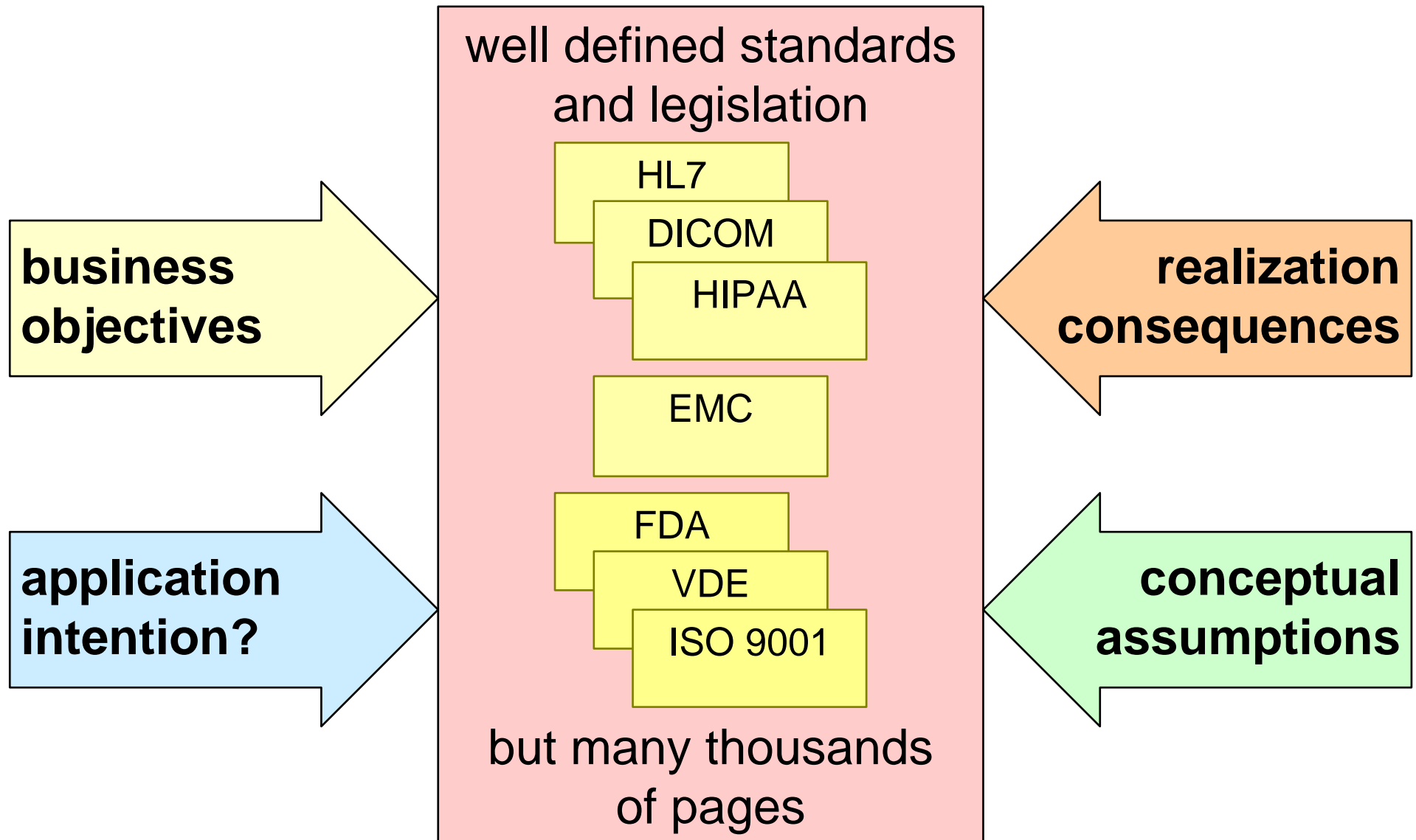
ny: 16 bit unsigned integer

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

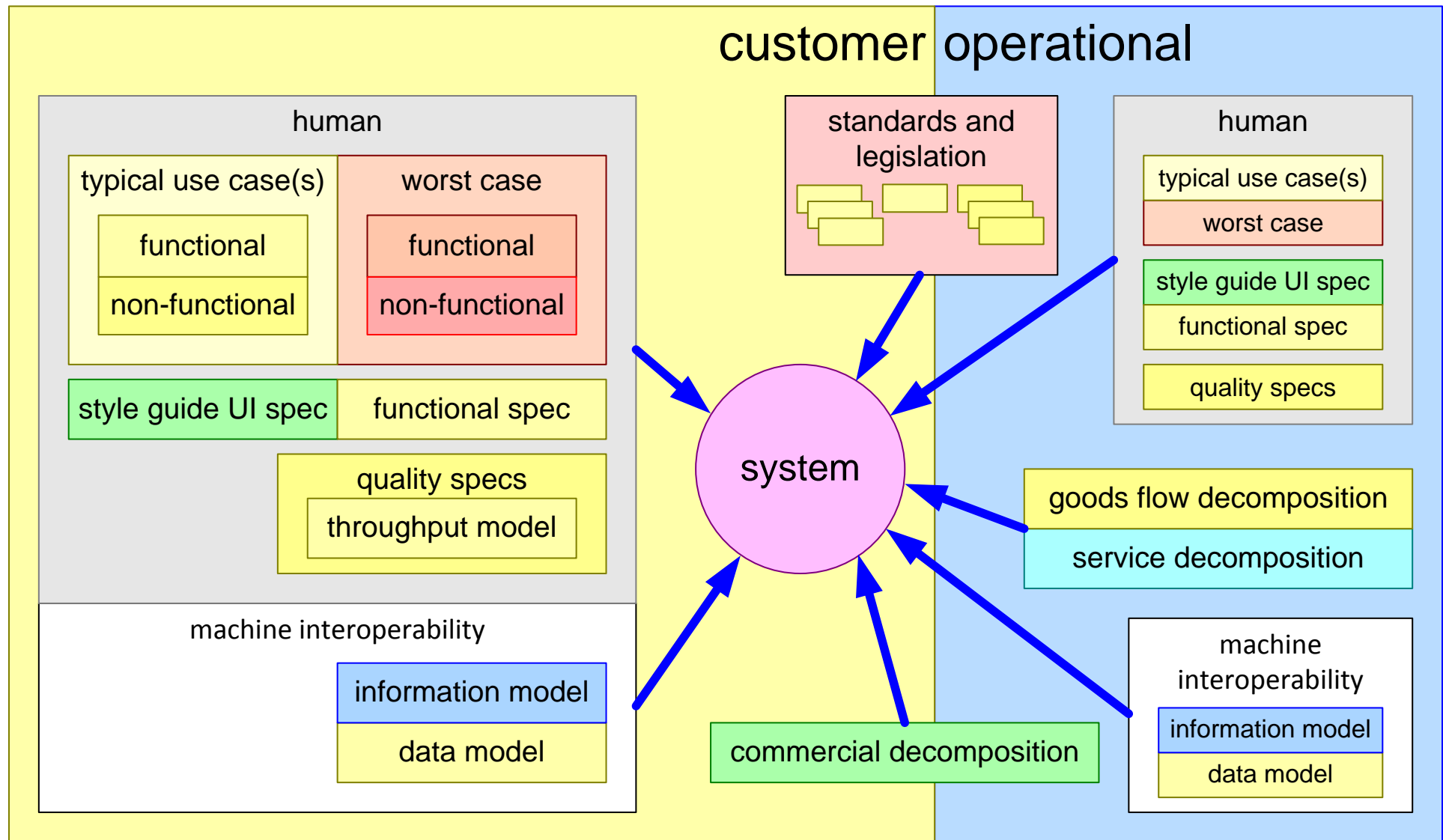
Example of performance modelling



The role of standards



Functional view summary



Functional view = What: externally observable