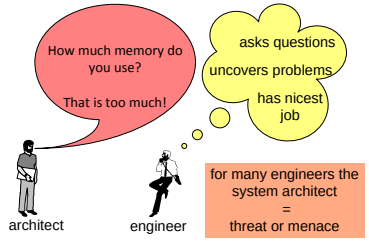
The System Architect; Meddler or Hero?

by *Gerrit Muller* University of South-Eastern Norway-NISE
e-mail: gaudisite@gmail.com
www.gaudisite.nl

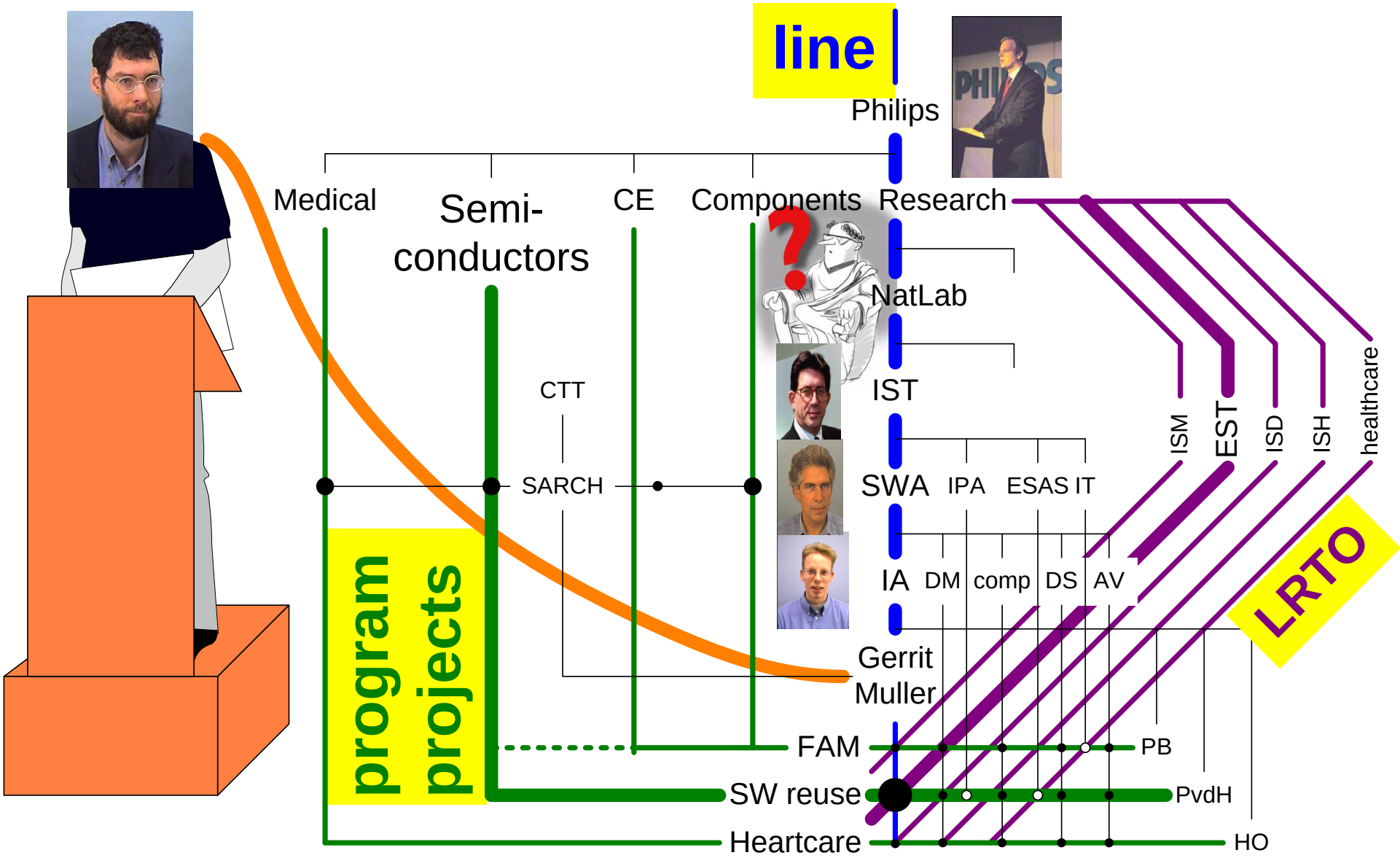
Abstract

Describes architecting and the task of the architect, with emphasis on bridging the **why**, **what** and **how** of a product. The memory usage of a medical workstation is used as practical illustration.
The introduction of a system architect in an architecture unaware organisation is described. A metamorphosis takes place from a threatening meddler into an appreciated indispensable team member.

January 23, 2022
status: finished
version: 3.1



Where is Gerrit in multi-D organisation space?



Who is Gerrit? What is Gaudí ?

CT
OSS
NM
X-ray

MR

EasyVision

manager system engineering

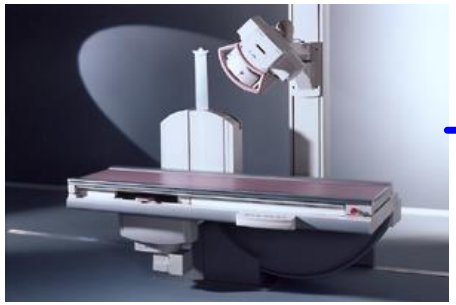
Gaudi

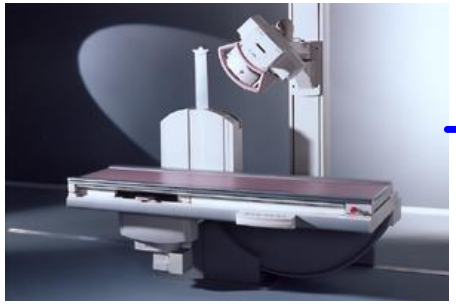
Philips Medical | ASML | Philips NatLab

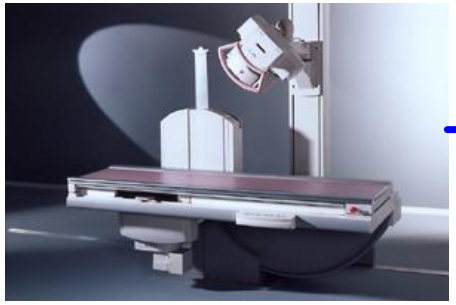
1980 1990 2000

Gaudi = consolidation education transfer + exploration + application consultation facilitation of system architecting

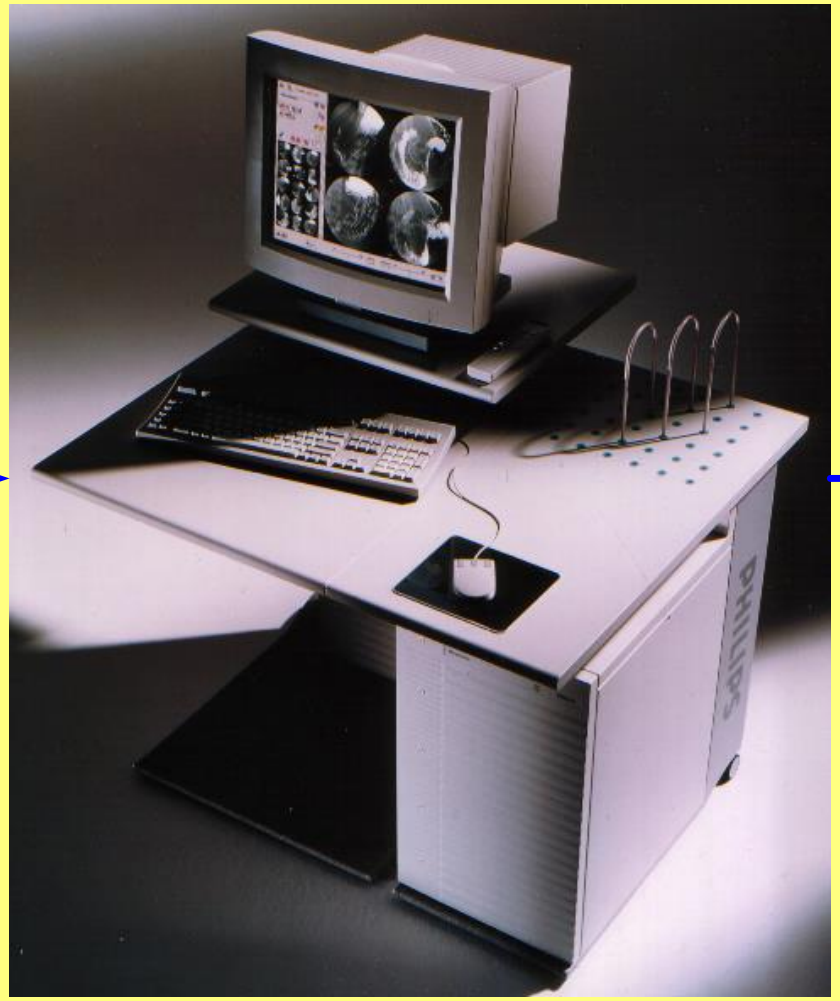
Practical illustration; a medical imaging workstation







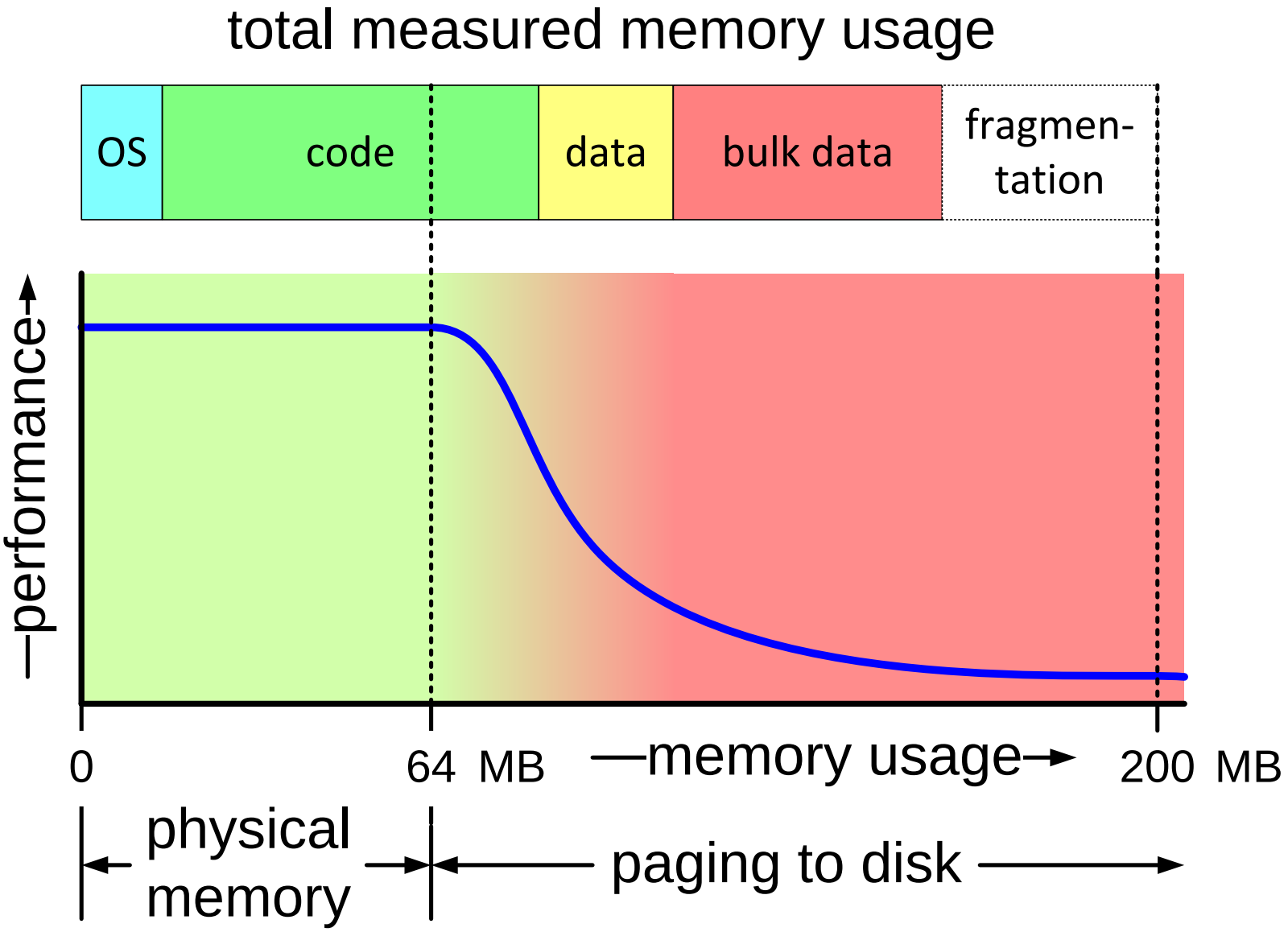
URF-systems



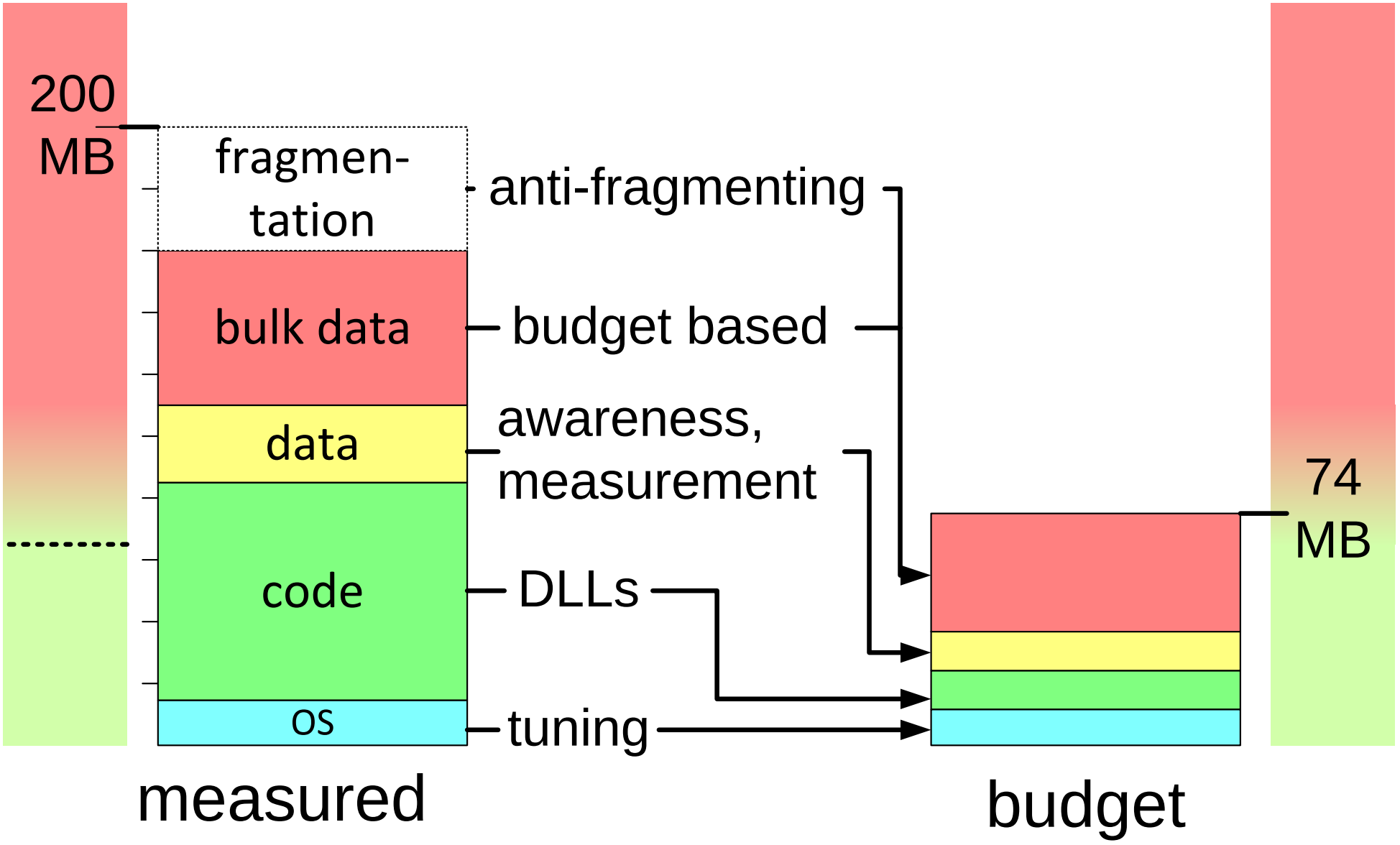
EasyVision: Medical Imaging Workstation

typical clinical image (intestines)

Problem: unlimited memory consumption (1992)



Solution: measure and iterative redesign



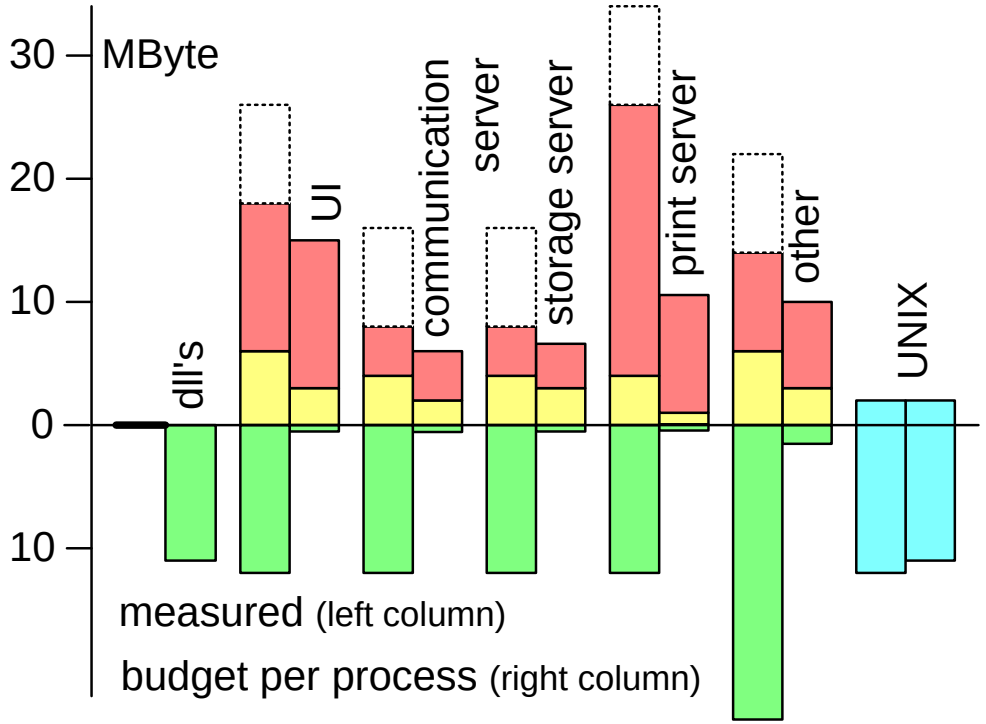
Method: budget per process

Budget:

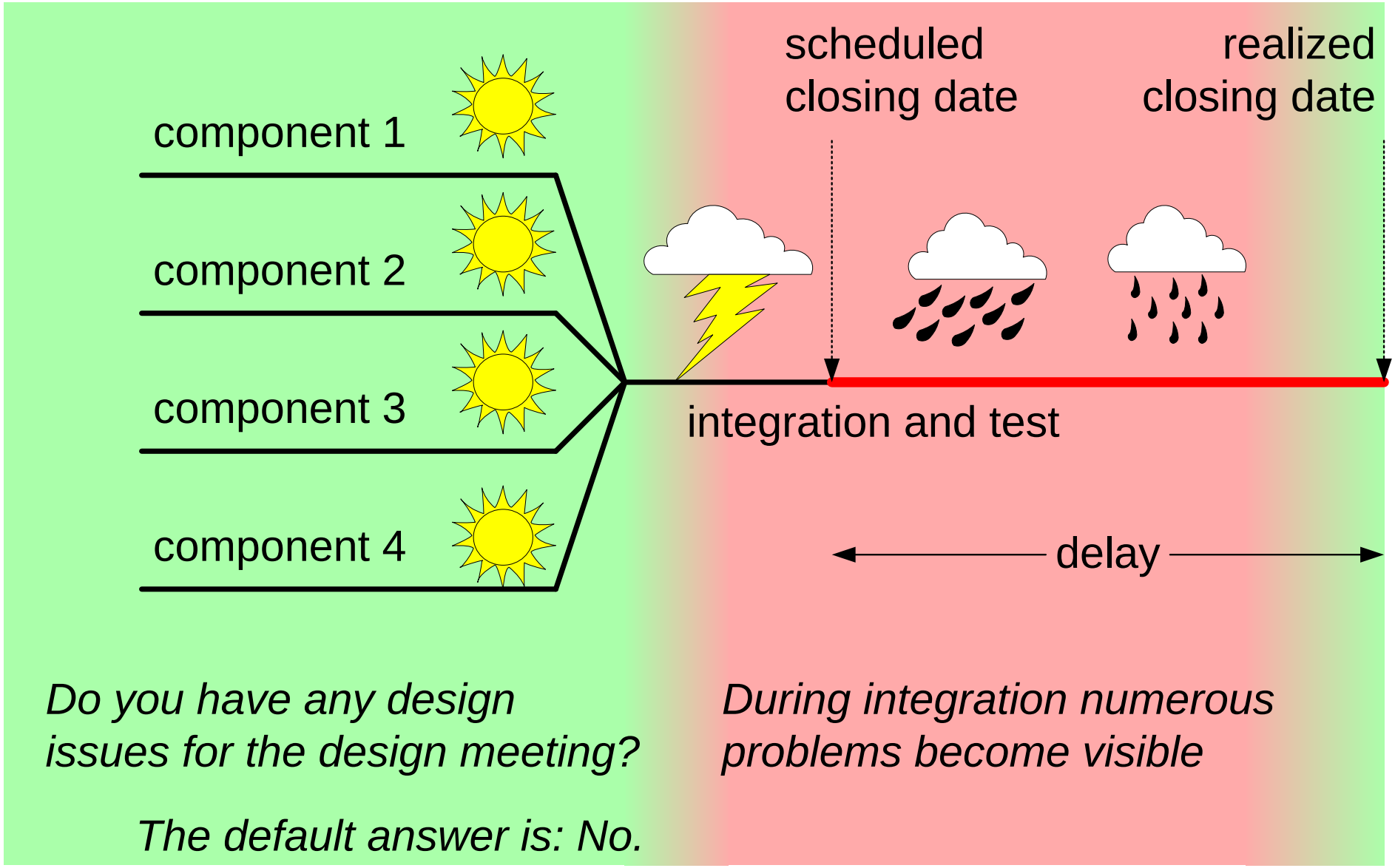
+ measurable

+ fine enough to provide direction

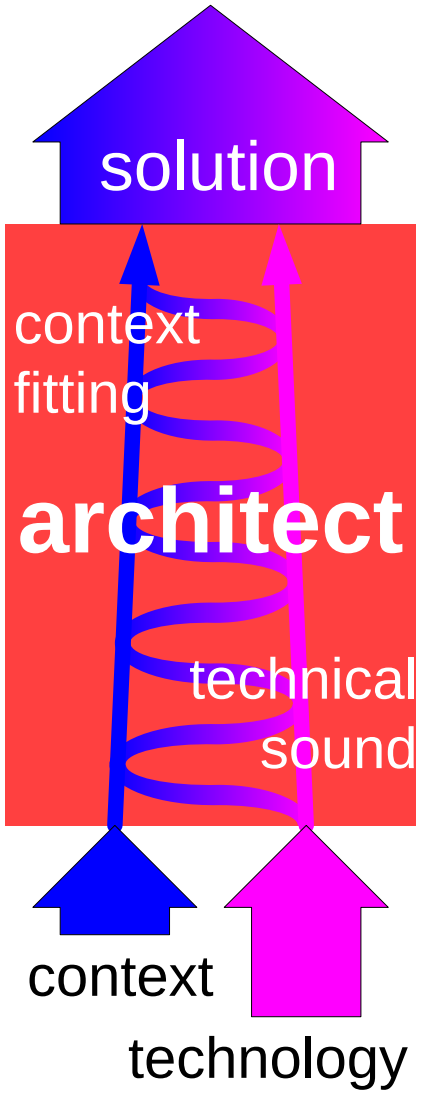
+ coarse enough to be maintainable



Integration uncovers hidden problems



Architecting scope

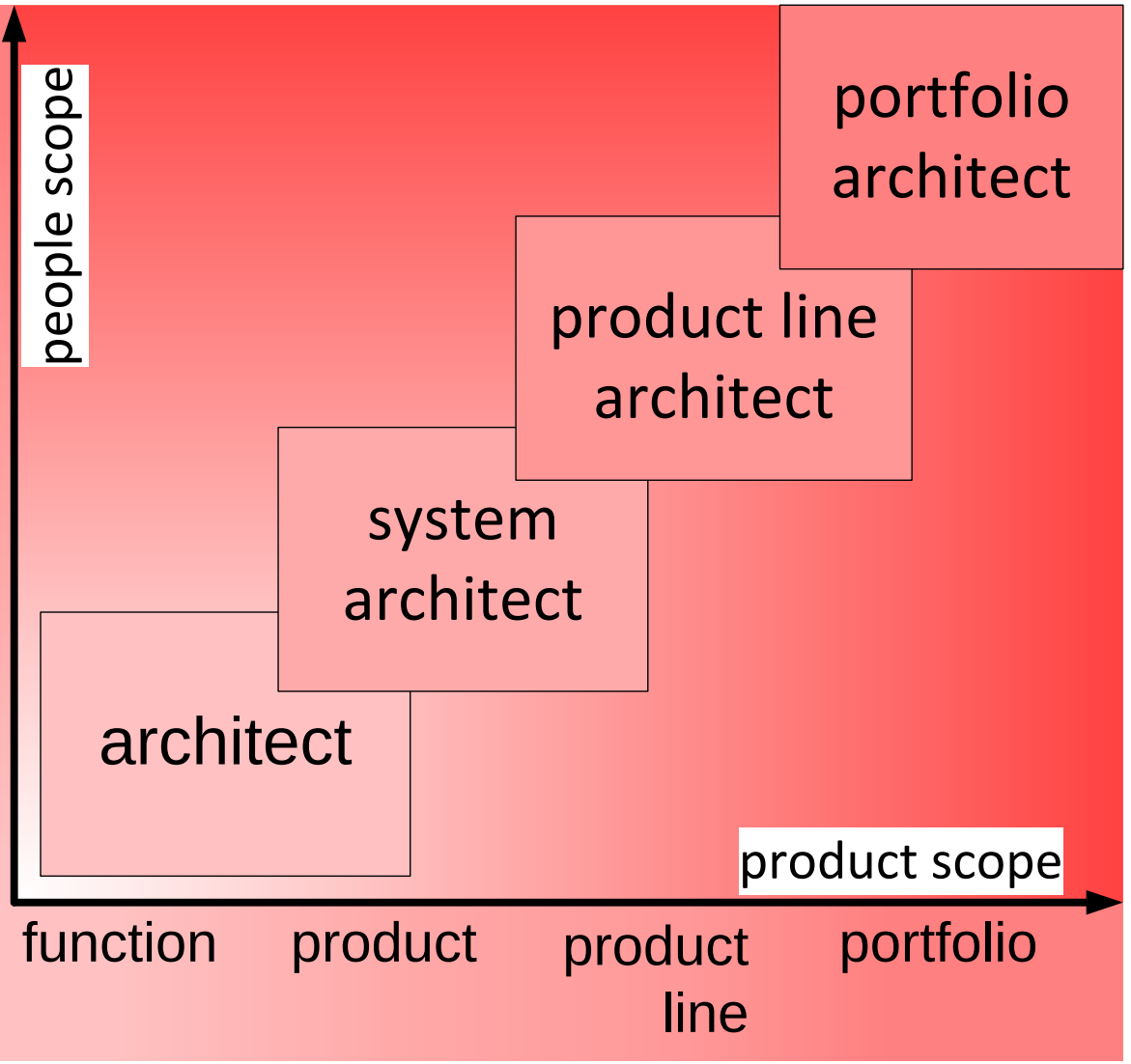


including
individuals
(human factors)

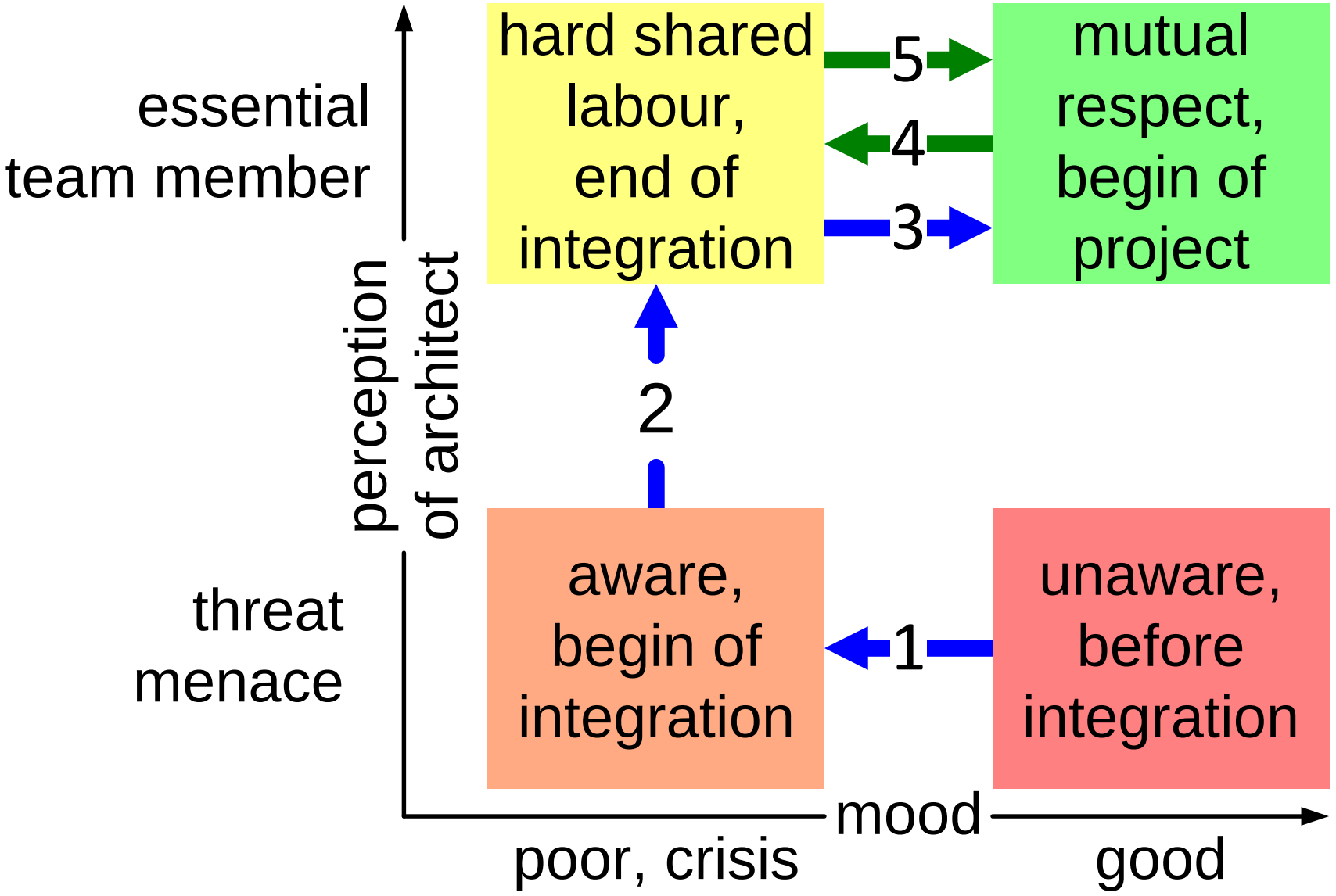
including
stakeholders

including
designers
(process)

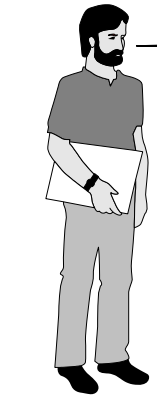
technology
only



Architecture awareness



The engineer's perception of the architect



How much memory do you use?

That is too much!

architect

asks questions

uncovers problems

has nicest job

engineer

for many engineers the system architect
=
threat or menace

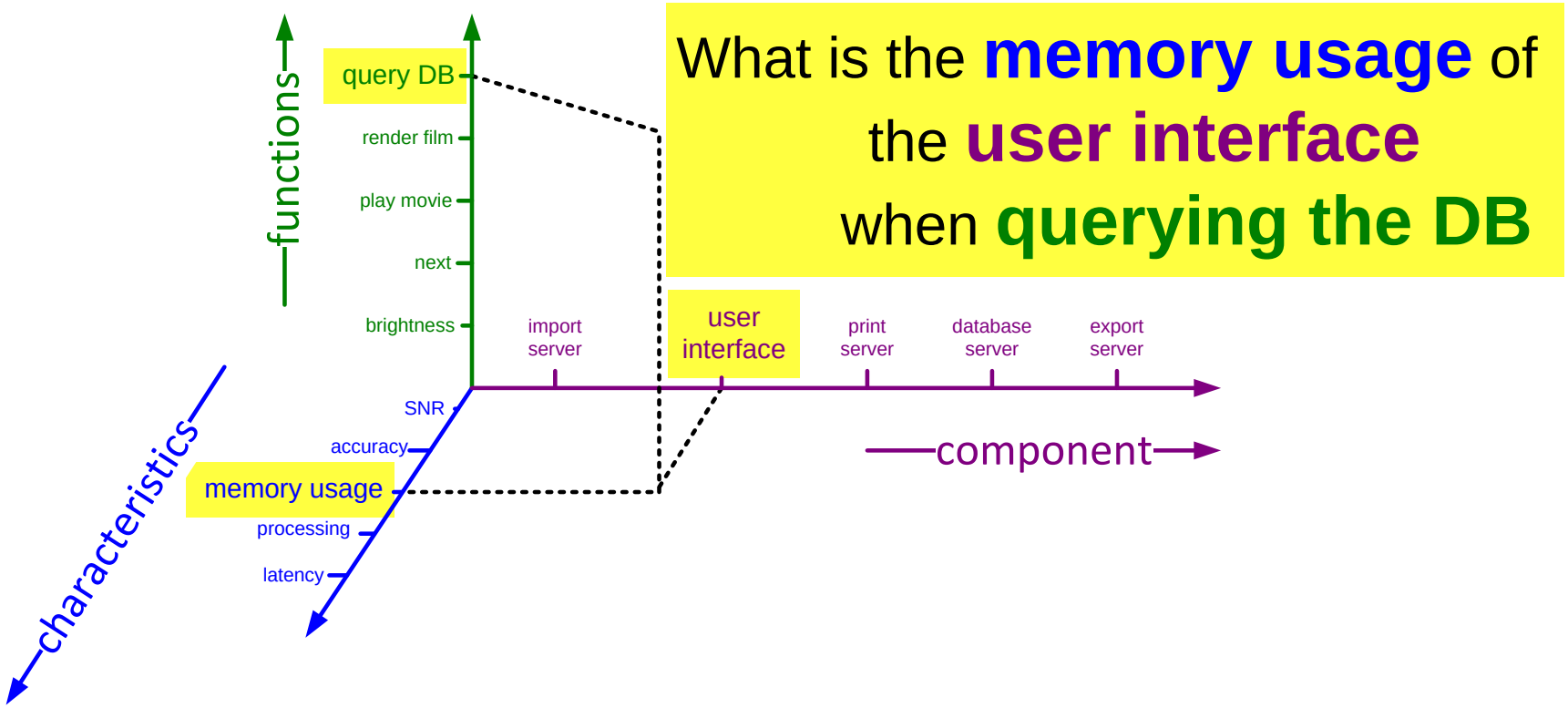
Understand the customer to drive design choices

What does Customer need in Product and **Why**?

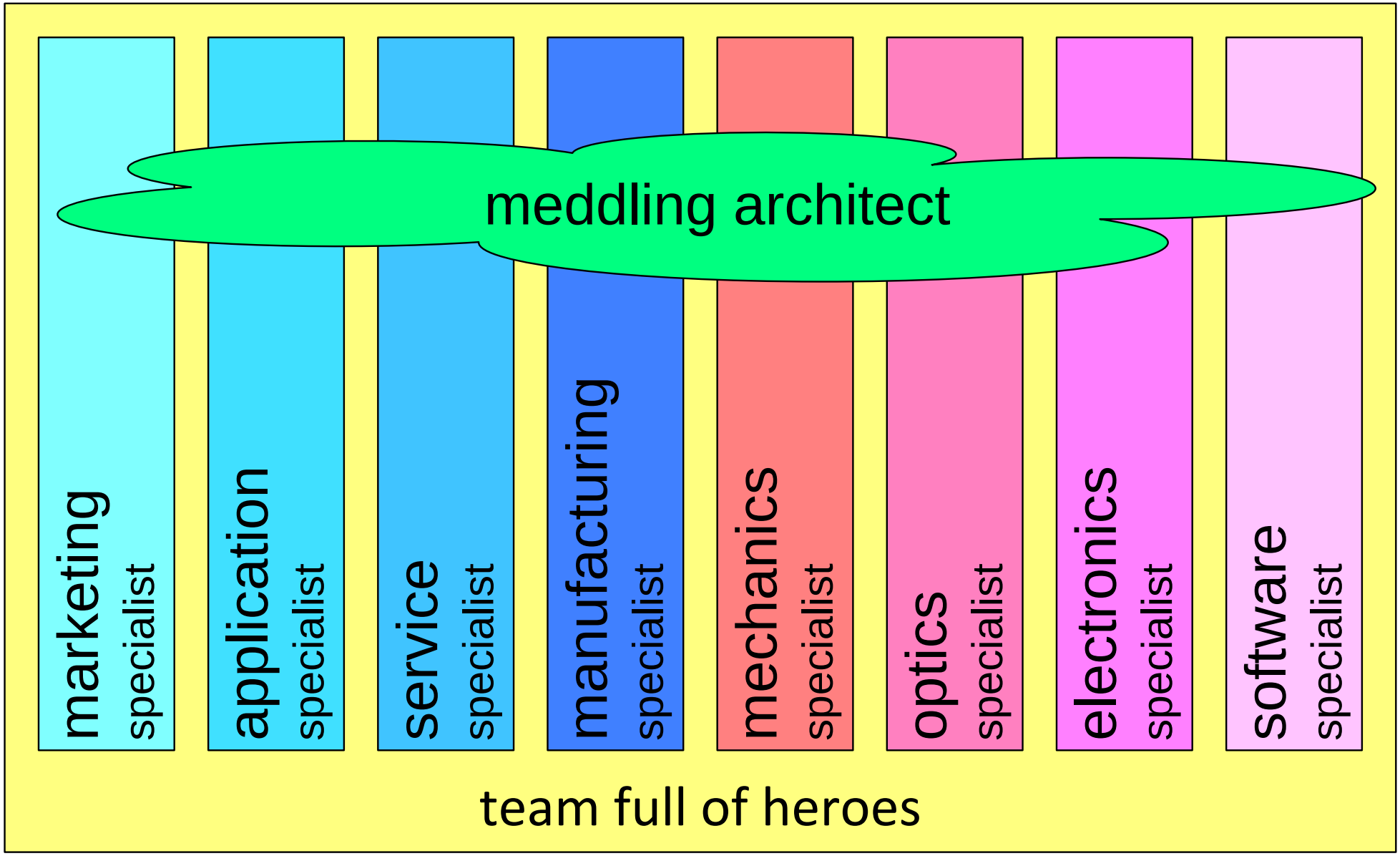
Customer **What**	Customer **How**	Product **What**	Product **How**	
Customer objectives	**A**pplication	**F**unctional	**C**onceptual	**R**ealisation
key drivers: * throughput * diagnostic quality * film saving - street price 50k$	* 3 exam rooms per exam: * 20 images * auto-print on 3 sheets		decomposition anti-fragmentation DLL's	measured 200 MB budget 74 MB

Method to detect problems: Question generator

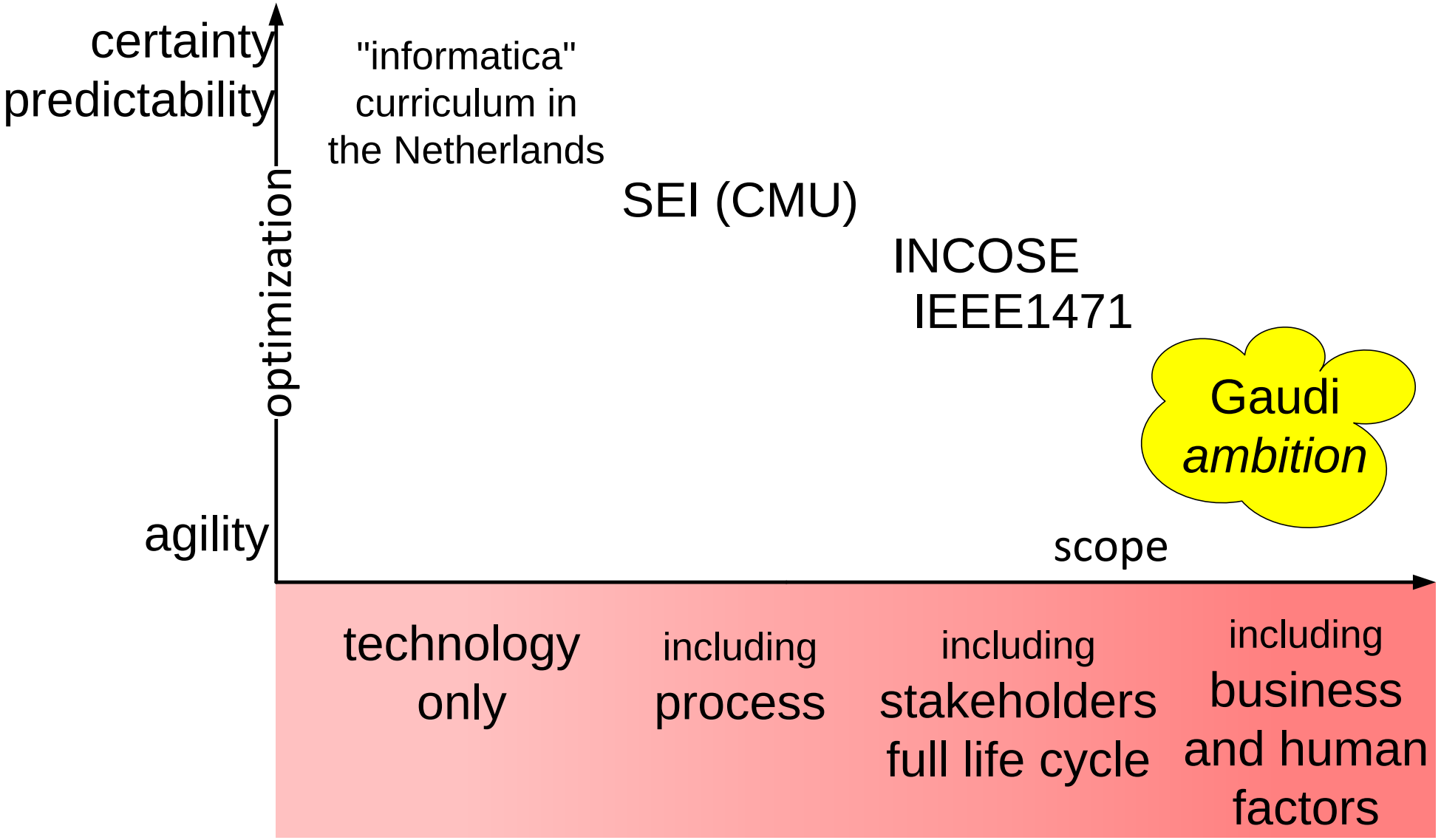
How about the **<characteristic>**
of the **<component>**
when performing **<function>**?



Conclusion


meddling architect
marketing specialist
application specialist
service specialist
manufacturing specialist
mechanics specialist
optics specialist
electronics specialist
software specialist
team full of heroes

Benchmarking



Some **A**cknowledgements

Hammer, Dieter
Hoogenstraaten, Wil
Mueller, Juergen
Gieles, Hans
Eggenhuisen, Huib
Kloprogge, Raymond
Engel, Bas
van Rijnsoever, Bart
Driesen, JGH
Schelkers, Raymond
van der Heijden, Jaap
Vermeulen, Gerry
Stroucken, Marc
Wijnstra, Jan Gerben
Algra, Egbert
Derks, Frans
Faber, Albert
Aarts, Peter
Watabe, Yasuma
van Ouwerkerk, H
Huijnen, Ton
Gonot, Mathieu
van Splunter, Andre
van Rooijen, Joost
Verberkt, Mark
van der Linden, Wim
Patrzalek, Jarek
Vergoossen, Theo
Gijsbers, Rob
Huis in 't Veld, Robert
Joosten, Jan
Mulder, Alwin
de Wit, Paul
Poesse, Jan
Spaak, Wim
Thus, Frank
van Velden, Jeroen
van Venrooy, Roland
Dobbelsteen, Jan
de Waal, Klaas
Muijen, M
Peters, Jo
van Bommel, Pieter Jan
Thijssen, Henk
Boot, Gert Jan
Vullings, Erik
Vermeer, Ad
Peeters, Bob
Obbink, Henk
Bas, Han
Rankers, Adrian
Akiwumi-Assani, Olu
Gopalan, Rajaraman
Misdom, Han
Schatorie, John
Boer, Richard
Penners, Maurice
America, Pierre
Jaspers, Peter
Versteijlen, Joost
Beelen, Peter
Blijd, Jarl
Dijkema, Marcel
Roelandt, Werner
Janson, Paul
Bandakka, Mahesh
Ledeboer, Jodie
Geron, Nic
Zieringer, Peter
Beuk, Leo
Koolen, Gertjan
Koushik, Sudeendra
Milosevski, Vlatko
van den Broek, Ger
de Kruif, Peter
Daenen, Steven
Soepenberg, Gerben
Bingley, Peter
Follon, Roel
Elzinga, Onno
van den Donker, Piet
Zwaans, Ben
Harmsze, Francoise
Jansen, Tom
van Gogh, Clemy
Wissink, Getty
Engelsma, Erwin
Stut, Wim
Luttikhuizen, Paul
Bruin, Jan
Gooren, Huub
den Dekker, Wim
van der Laak, Eric
Crins, Wim
Heerink, Lex
Schippers, Alef
Schreppers, Jurgen
Deckers, Robert
van Balen, Auke
Huiban, Cristian
van Loon, Gerard
van den Heuvel, Patrick
Lobo, Lincoln
van de Meulenhof, Dennis
Houtepen, Rob
Hofsink, Robert
Buurman, Hans
Zondag, Eddy
Veldmans, Ferdinand
Merkus, Paul
van Tuijl, Frank
Wouters, Kees
van der Sterren, William
Soede, Michiel
van Bommel, Luc
Krikhaar, Rene
van den Brink, Johan
Ham, Kees
Bos, Erik
Pijpers, Frank
Medema, Jeroen
Kaag, Bjorn
Giesselman, Timo
Vos, Frans
de Greef, Pierre
Fischer, Stefan
Pu, Xuemei
Boom, Sjirk
ten Pierick, Henk
Stroucken, Louis
Young Tai Liu
van der Steen, Marcel
Siereveld, Ad
van Bakel, Gerian
Engbers, Rene
van Wetten, Frank
Stevers, Frank
Wubben, Rob
Schellingerhout, Nico
Vugts, John

Gaudí homepage

http://nlww.natlab.research.philips.com:8080/research/swa_group/muller/

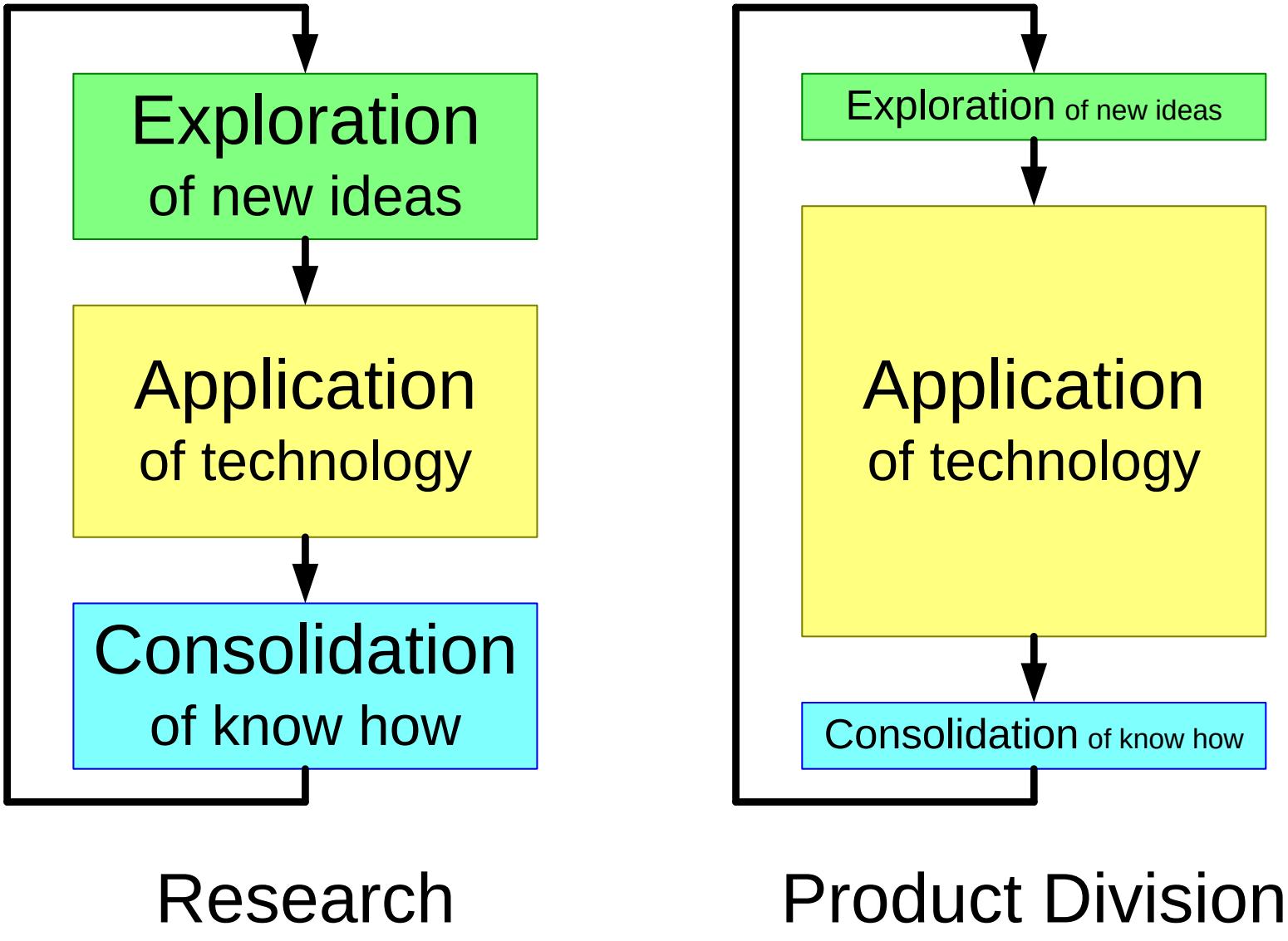
containing:
more than 30 recent articles and or presentations
course material SARCH, ESA stakeholders, OOTI req eng

links on this homepage:

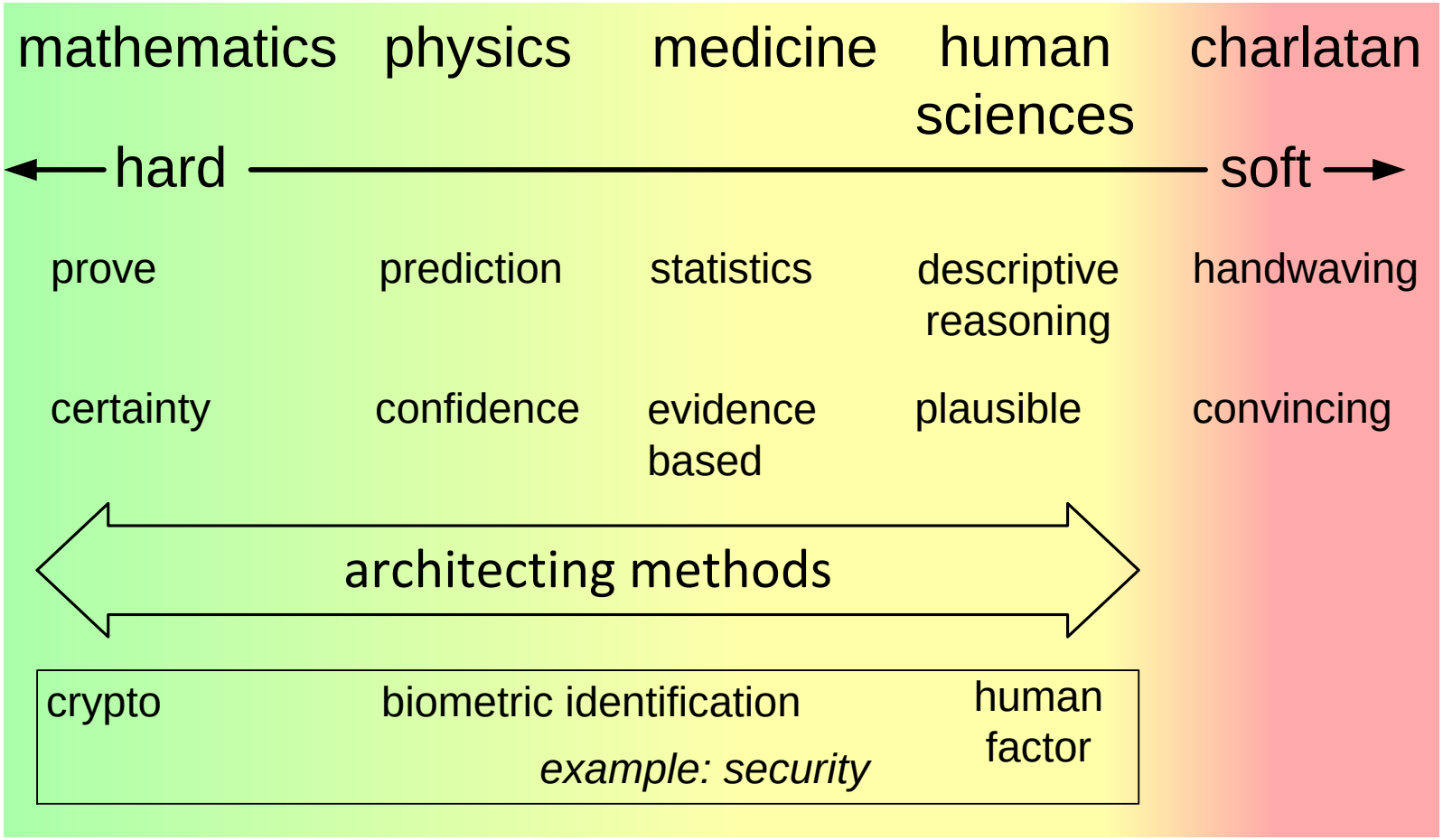
this presentation	MeddlerOrSaviorSlides.pdf
annotated text	MeddlerOrSaviorPaper.pdf
background information Medical Imaging	MedicalImagingPaper.pdf
original documentation Medical Imaging	oldPresentations.html

Answers to frequently asked questions

Architecture and research? The technology management cycle



Is architecting scientific?



legend

hard science

soft science

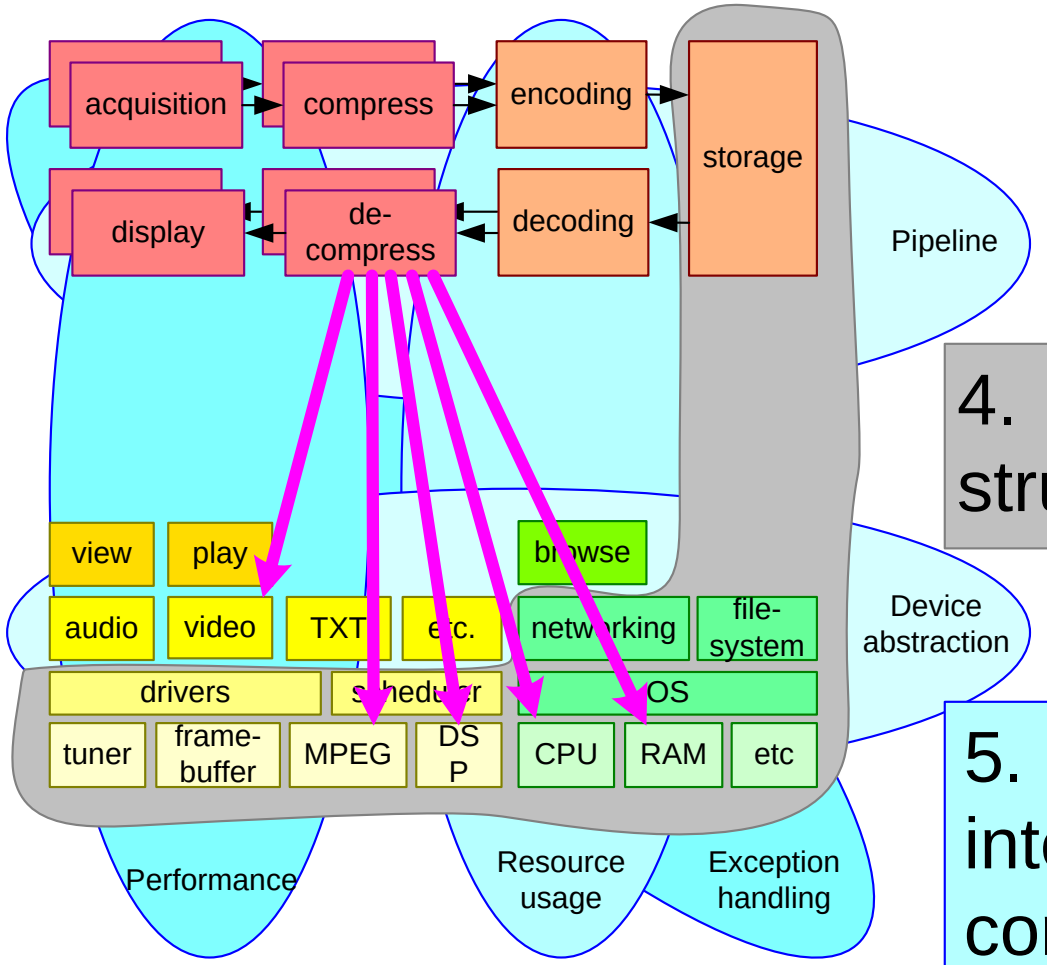
no science

Guiding how

1. Functional Decomposition

3. Allocation

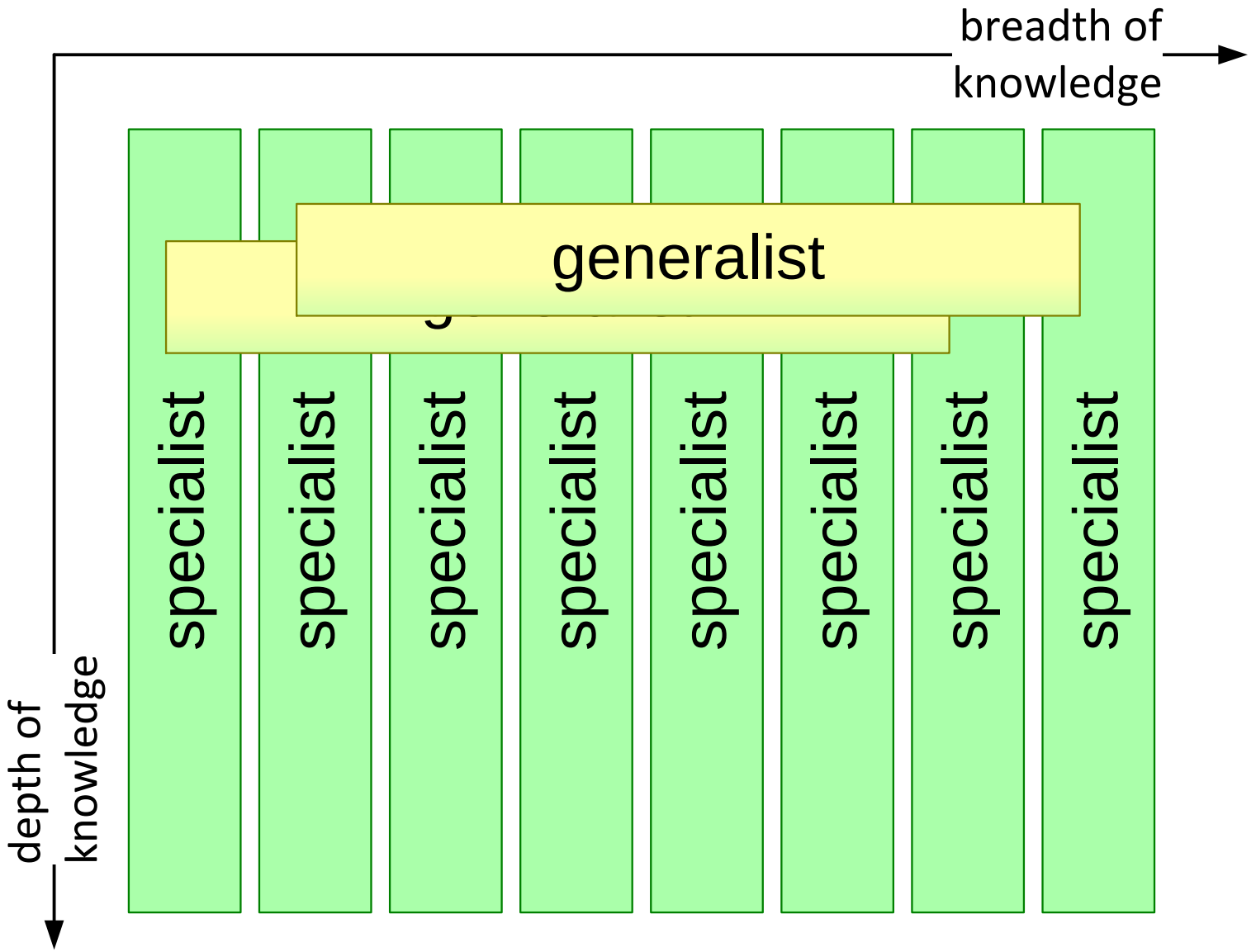
2. Construction Decomposition



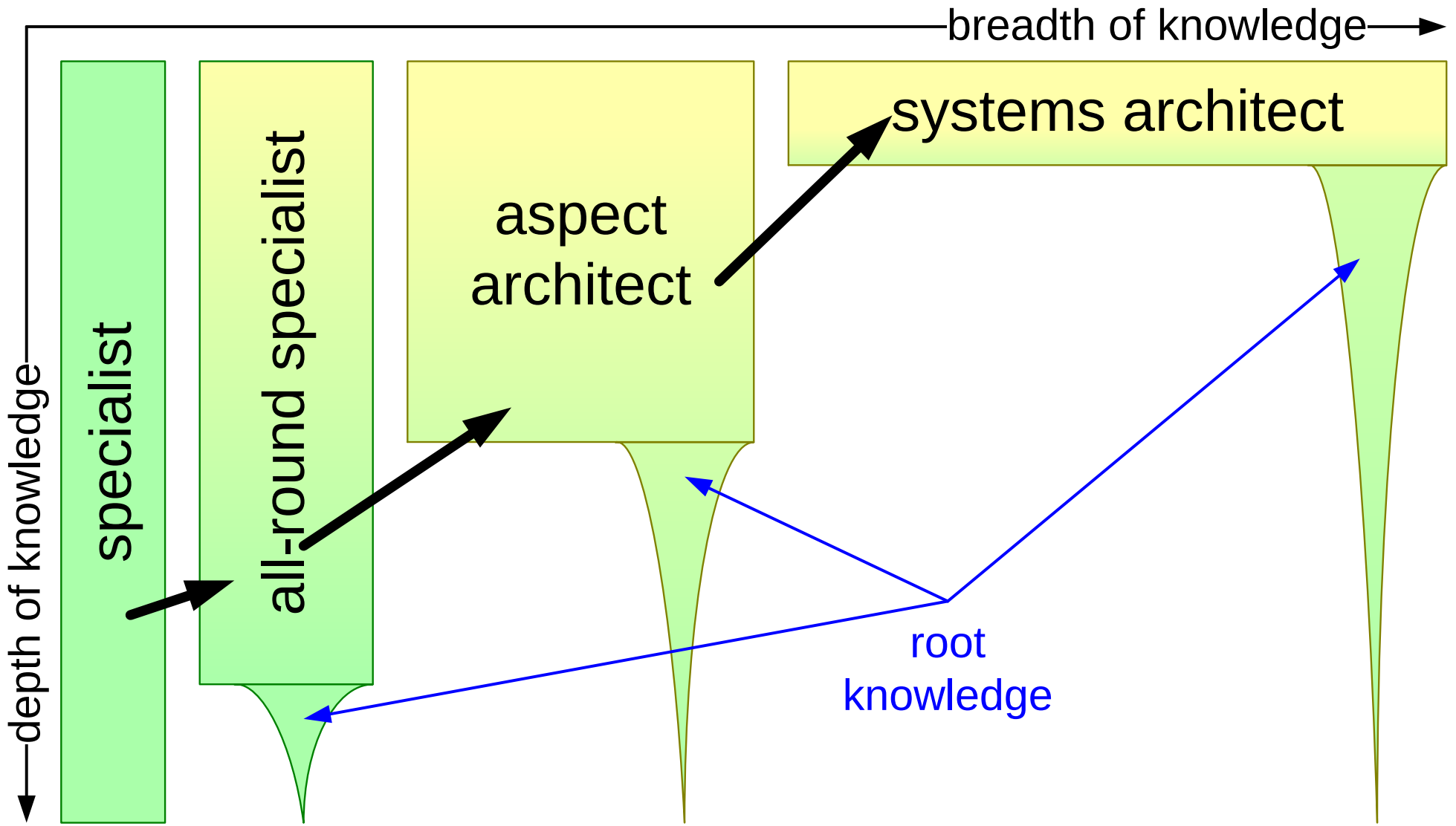
4. Infra-structure

5. Choice of integrating concepts

The architect as integrator

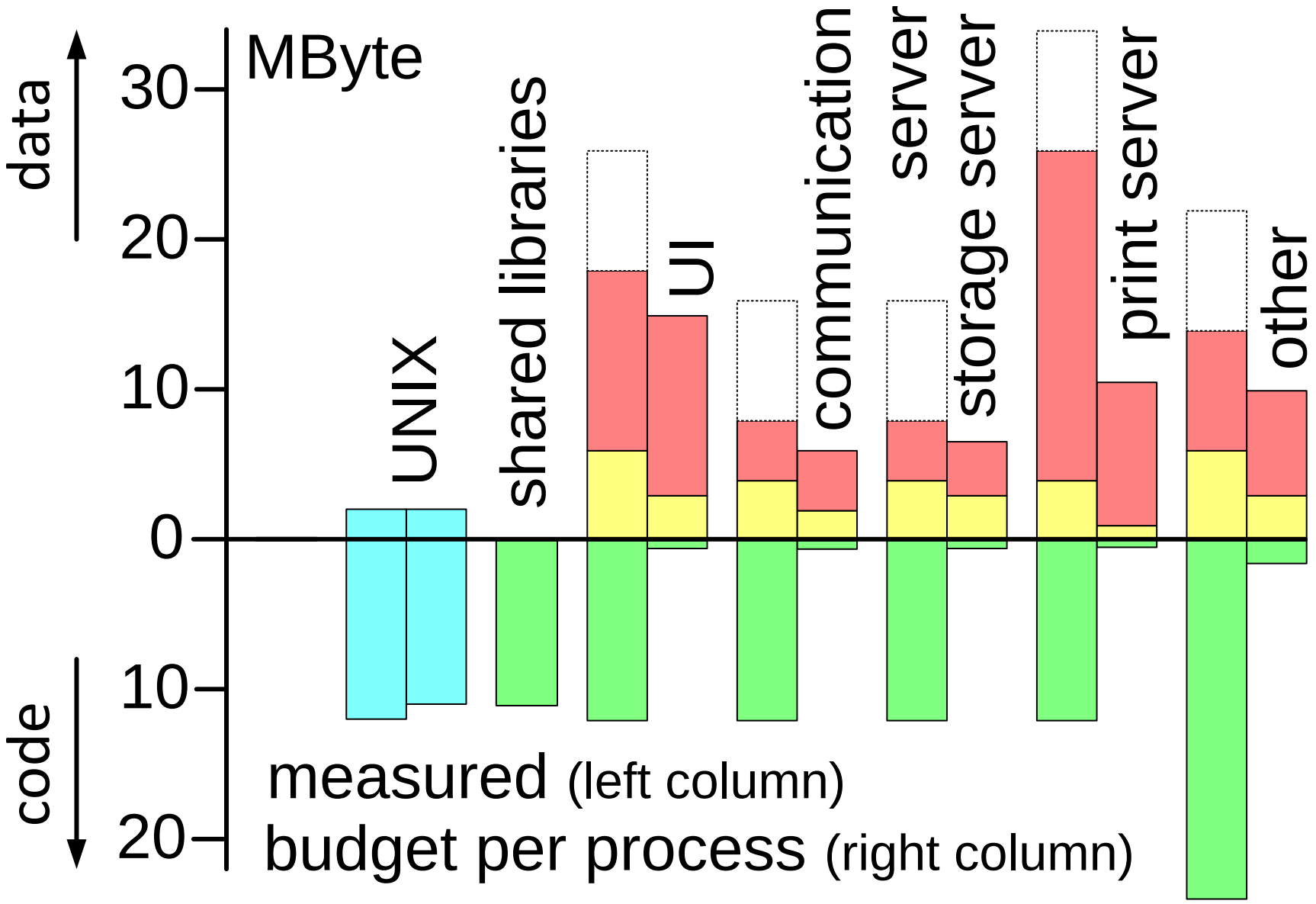


The architect maintains technical roots



sheets that didn’t make it into the final presentation

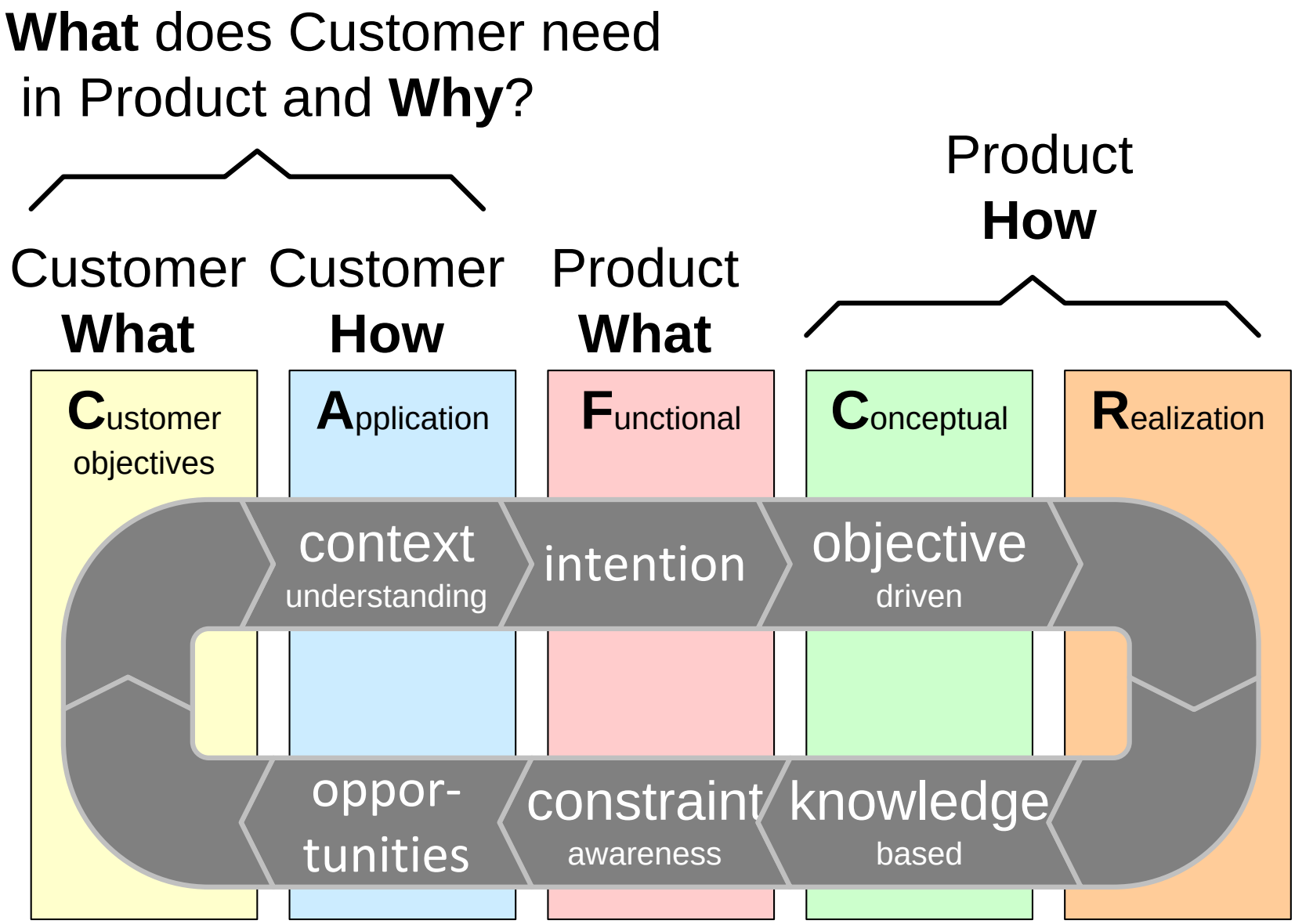
Memory Budget; too detailed



Engineers perception; too detailed, too much text

Engineers feeling caused by	architect's activity	illustrated by examples of architect behavior
Which can be nasty, if you don't have any answer	- Asks questions	How much memory do you use ?
Or worse, a problem is hiding underneath	- Identifies risks and problems	Your functions together use more memory than provided
A judgement is given	- Voices opinions	That is too much
Meddling in my backyard, without any substantial know how	- Does have ***holes*** in know how	Takes decisions without knowing how memory intensive a hashed dictionary is
While we are busy with detailed bug fixing and maintenance	- Does the nice work	Makes a toplevel design, for instance a memory budget
Every question or problem explodes into an even more extensive set of questions and problems	- Does not provide definite answers or solutions	What happens if we have many small images? Or if we have very large images?

Integrating CAFCR; too glossy



Architect characteristics; too detailed, too much text

Engineers perception

- Asks questions
- Identifies risks and problems
- Voices opinions
- Does have ***holes*** in know how
- Does the nice work
- Does not provide definite answers or solutions

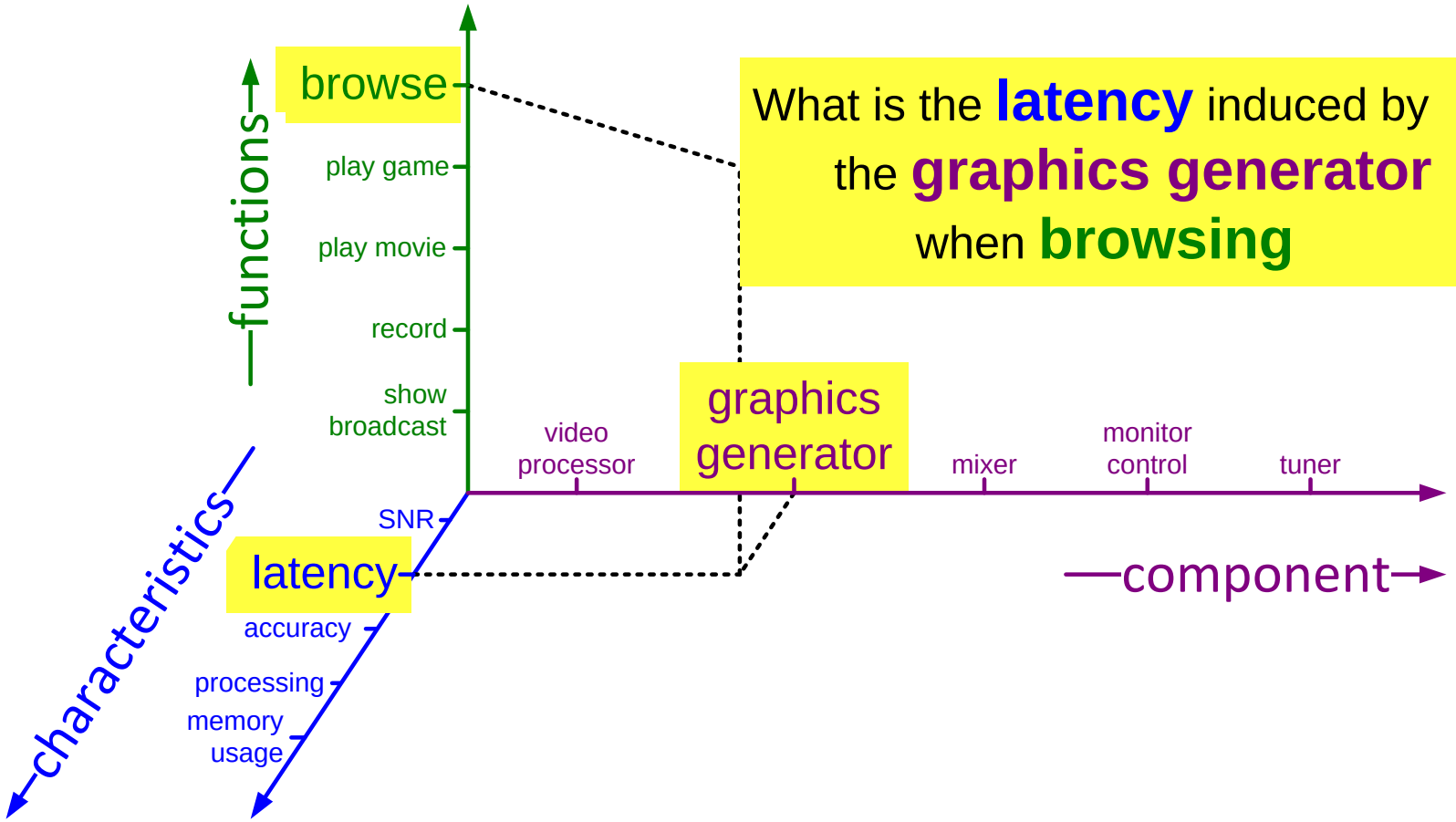
Required characteristic of the architect

Questions are the primary tool of an architect

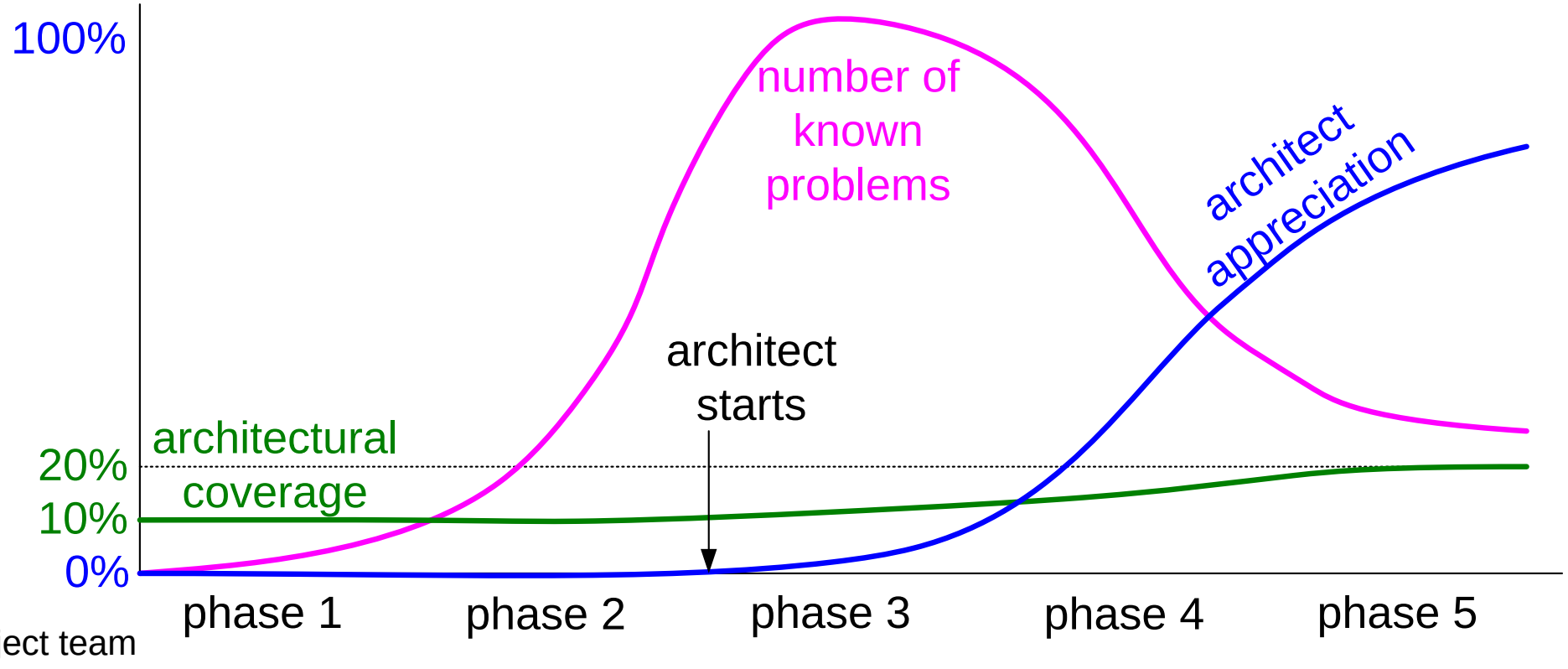
+ Prevention is valuable
+ Focus, selection is key, which requires vision and choices
+ Sometimes the architect needs to dive in deep
+ There is more nice work needed than we ever can do
+ Keeps repeating the previous actions

Question generator; other domain

How about the **<characteristic>**
of the **<component>**
when performing **<function>**?



Architecture Awareness Phases; too complex



project team	phase 1	phase 2	phase 3	phase 4	phase 5
mood	happy	problems	crisis	trouble-shooting	
architecture awareness	unaware	nagging awareness	threat	buils up credit	amazement