

# Light Weight Architecture revisited: the way of the future?

by *Gerrit Muller* Embedded Systems Institute  
e-mail: [gerrit.muller@embeddedsystems.nl](mailto:gerrit.muller@embeddedsystems.nl)  
[www.gaudisite.nl](http://www.gaudisite.nl)

## Abstract

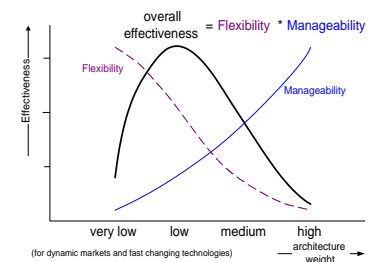
Technological developments change the consumer electronics market into a very dynamic market. CE manufacturers are used to realize product innovation by means of standardization, inside products as well as between products. Standardization and innovation are often conflicting activities. An approach is discussed to optimize the balance, based on "light-weight architectures".

The weight of an architecture determines how easy an architecture can be realized, changed and applied. An heavy architecture has many mandatory rules, which apply always and everywhere, with a large degree of detail. An heavy architecture provides a lot of certainties and control, but is more difficult to adapt to changing circumstances.

## 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.

February 10, 2011  
status: finished  
version: 0.0



# What is Architecture?

---

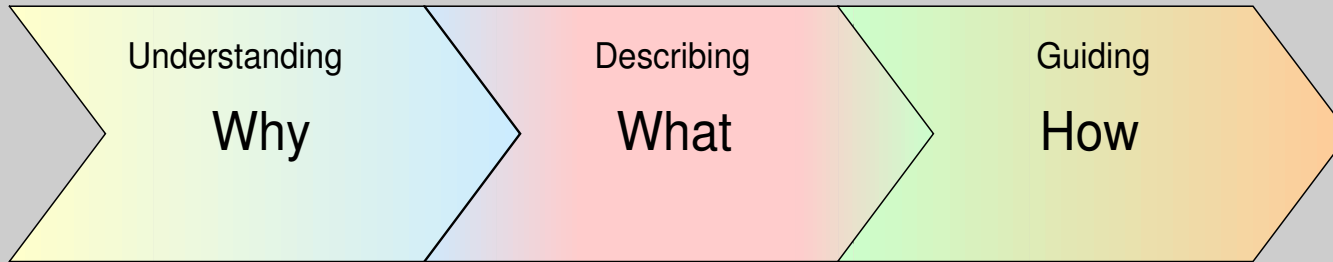


*Do the right things*

*Do the things right*

# Table of Contents

## 1. *Do the right things;* The Dynamic Market



## 2. *Do the things right;* Light-weight Architecture

This appliance may only be used for non commercial use  
accuracy +/- 200 g

On/Off



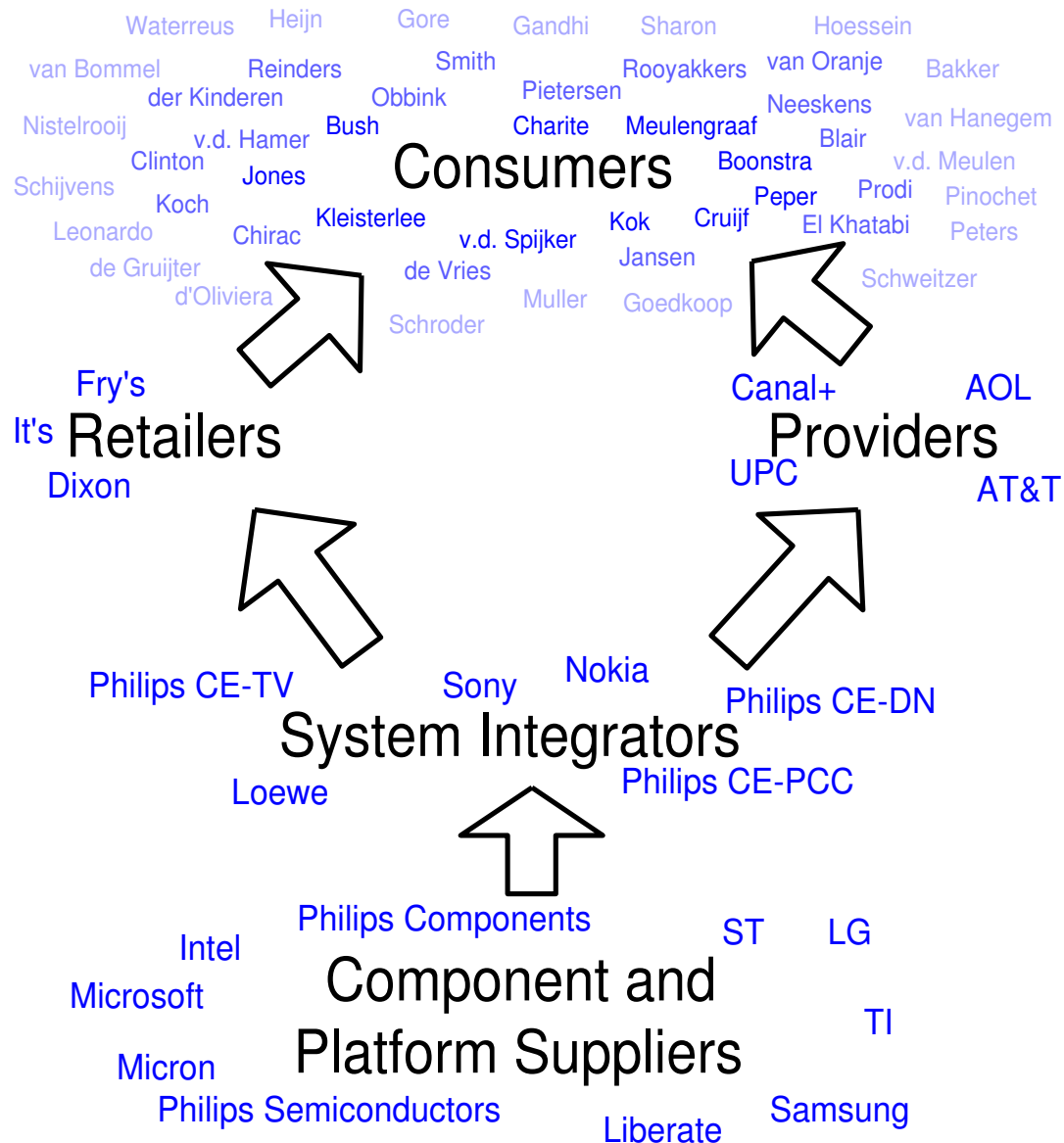
**25 Kg**

---

# Part 1:

## Do the right things; The Dynamic Market

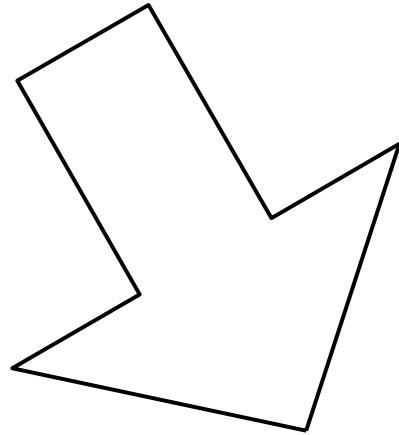
# Value chain



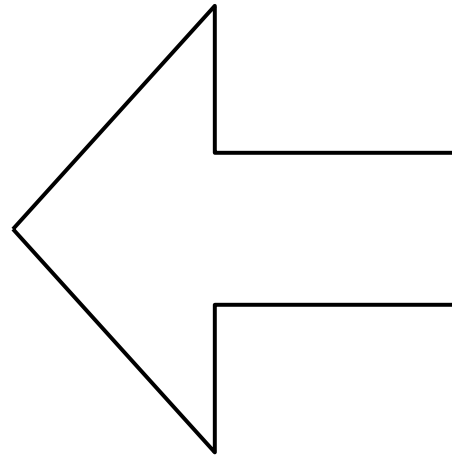
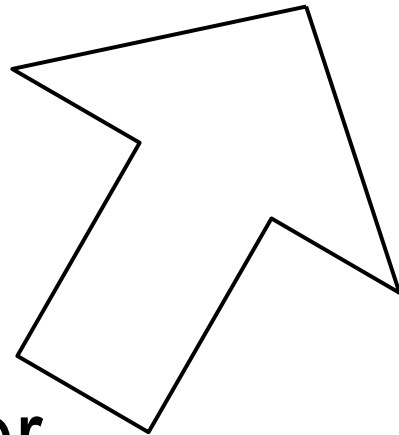
# Convergence

---

Telecom



Consumer



Computer

# Integration and Diversity



GSM phone



firewall



dvd



audio  
microset



pda



watch



sailboat



surveillance  
camera



cable  
modem



set top box



headphone



pen



garment



car



camera



speech



mp3



television



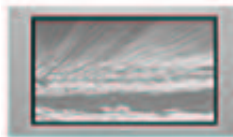
car navigation



computer



games



flat display



Communicator



Ambient Intelligence  
living room

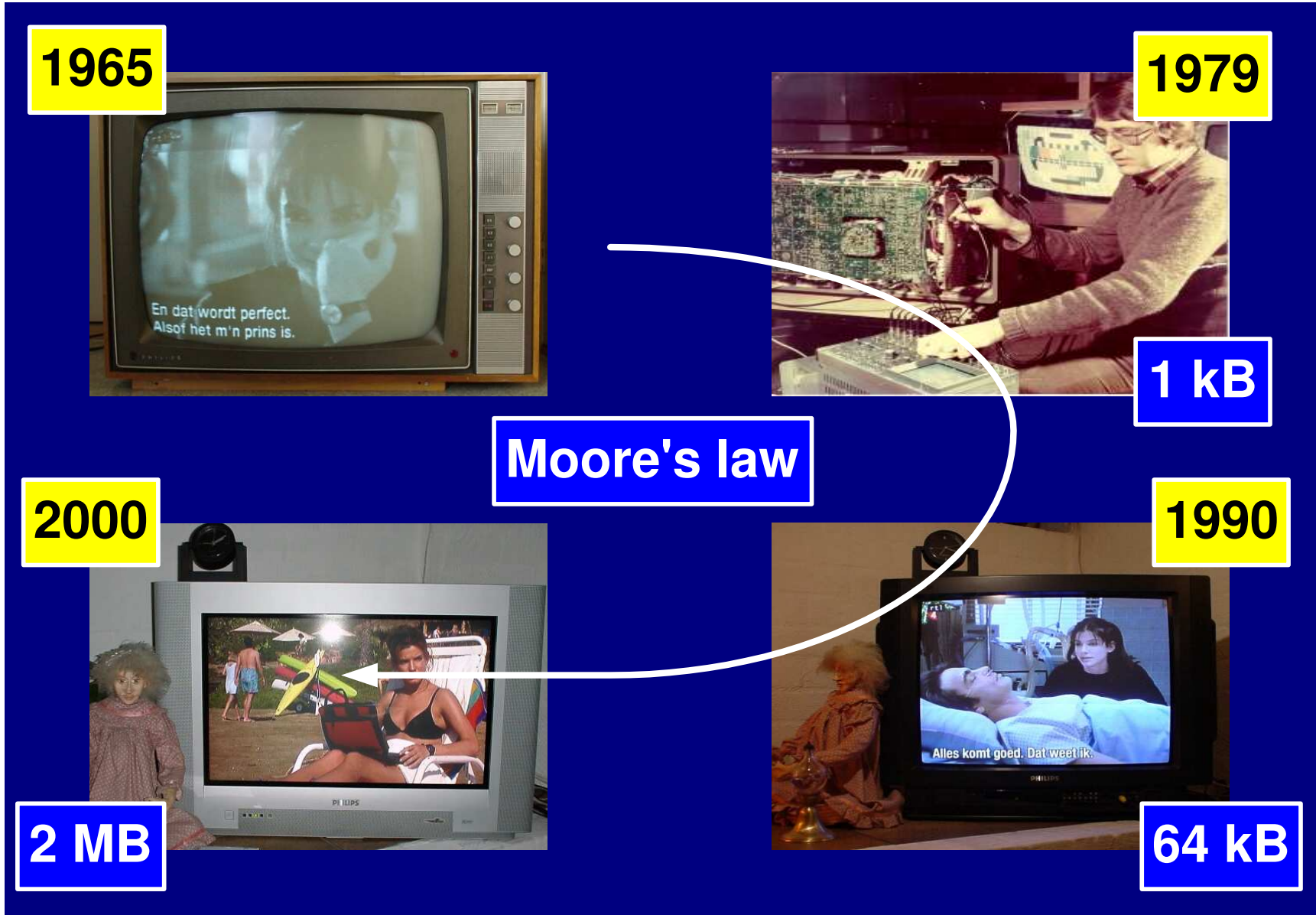
# Uncertainty (Dot.Com effect)



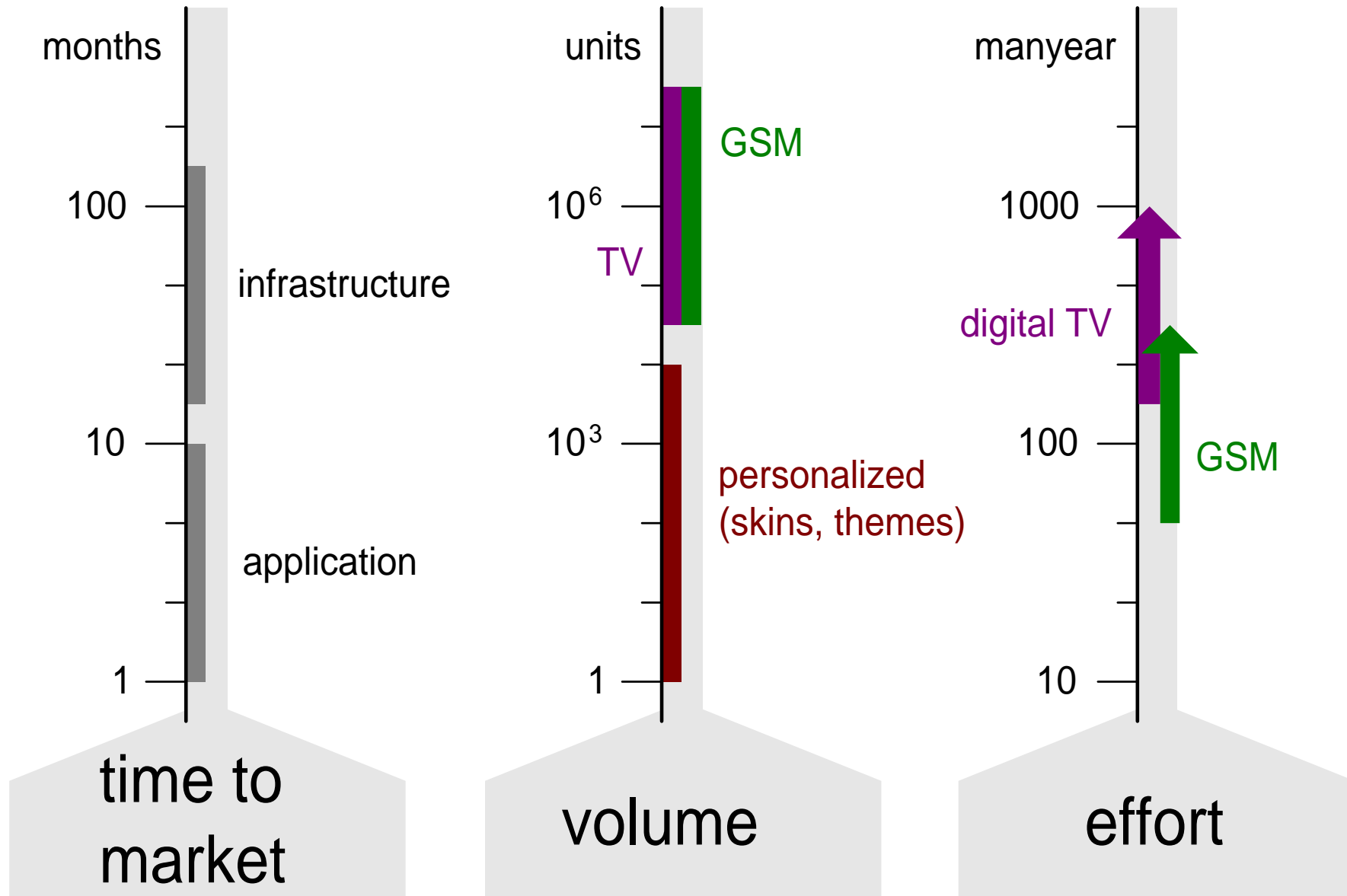
source: BigChart.com  
dd march 19, 2001

# Moore's law

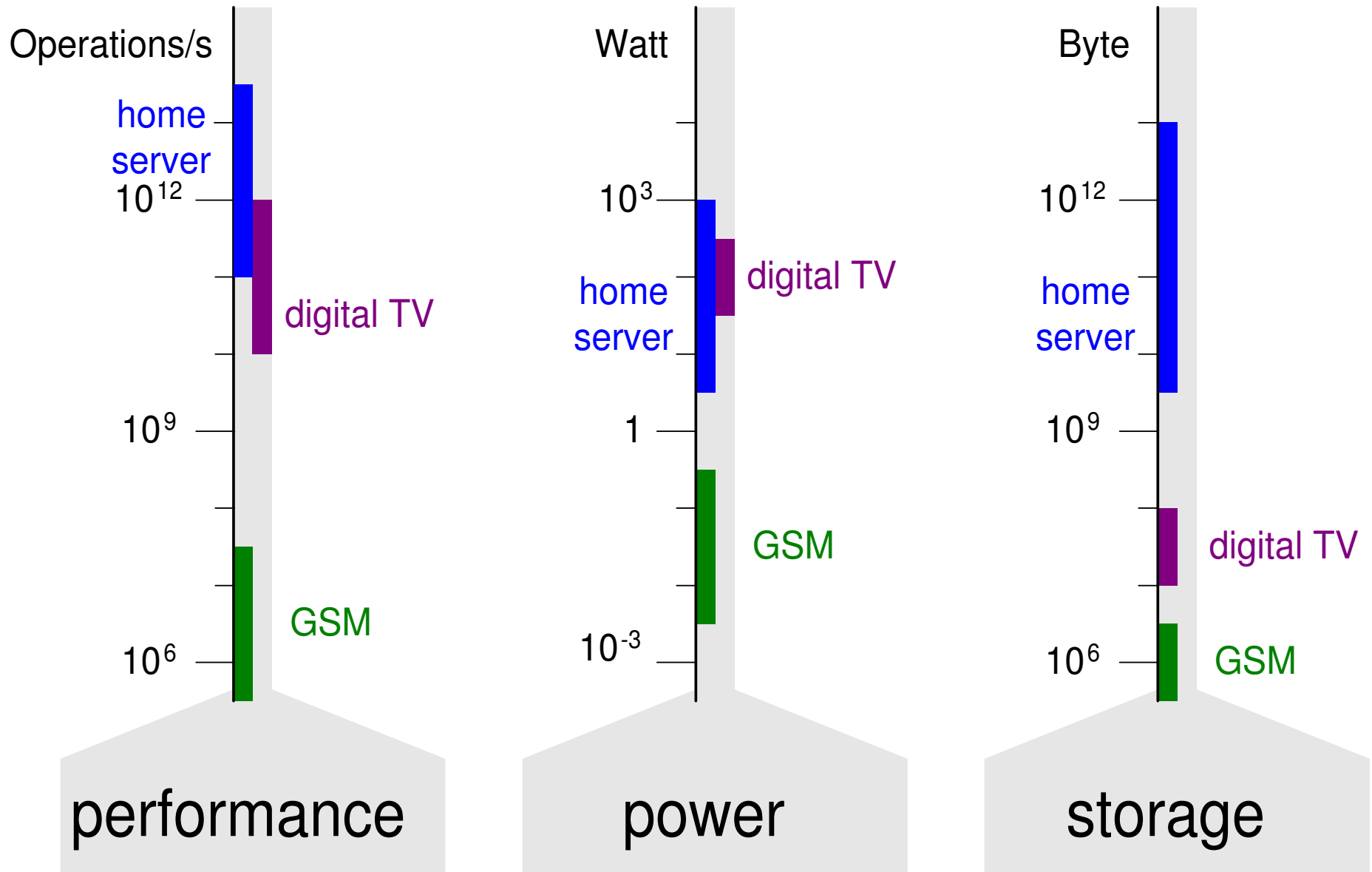
From: COPA tutorial, Rob van Ommering



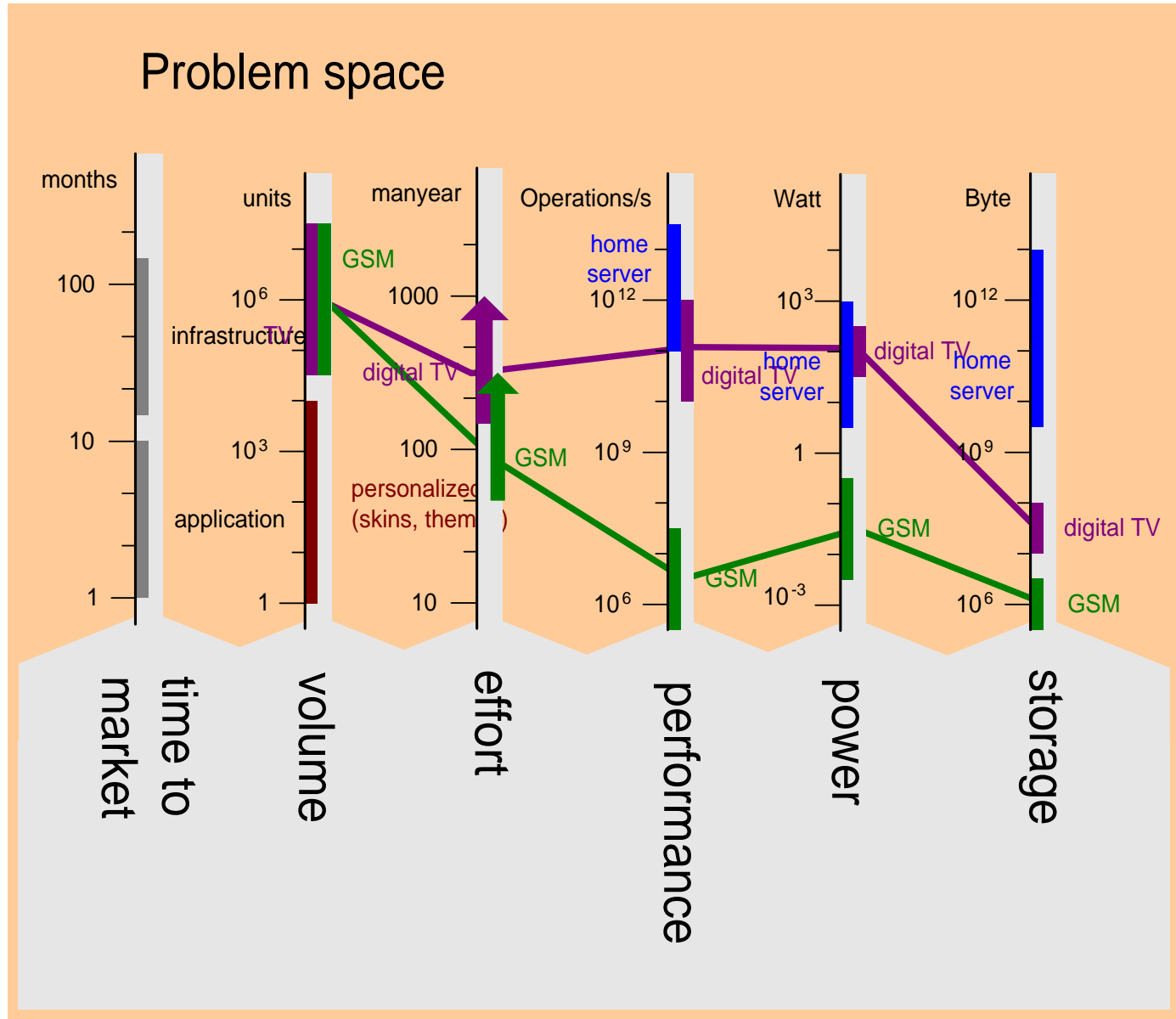
# System Integrator Problem Space - Business



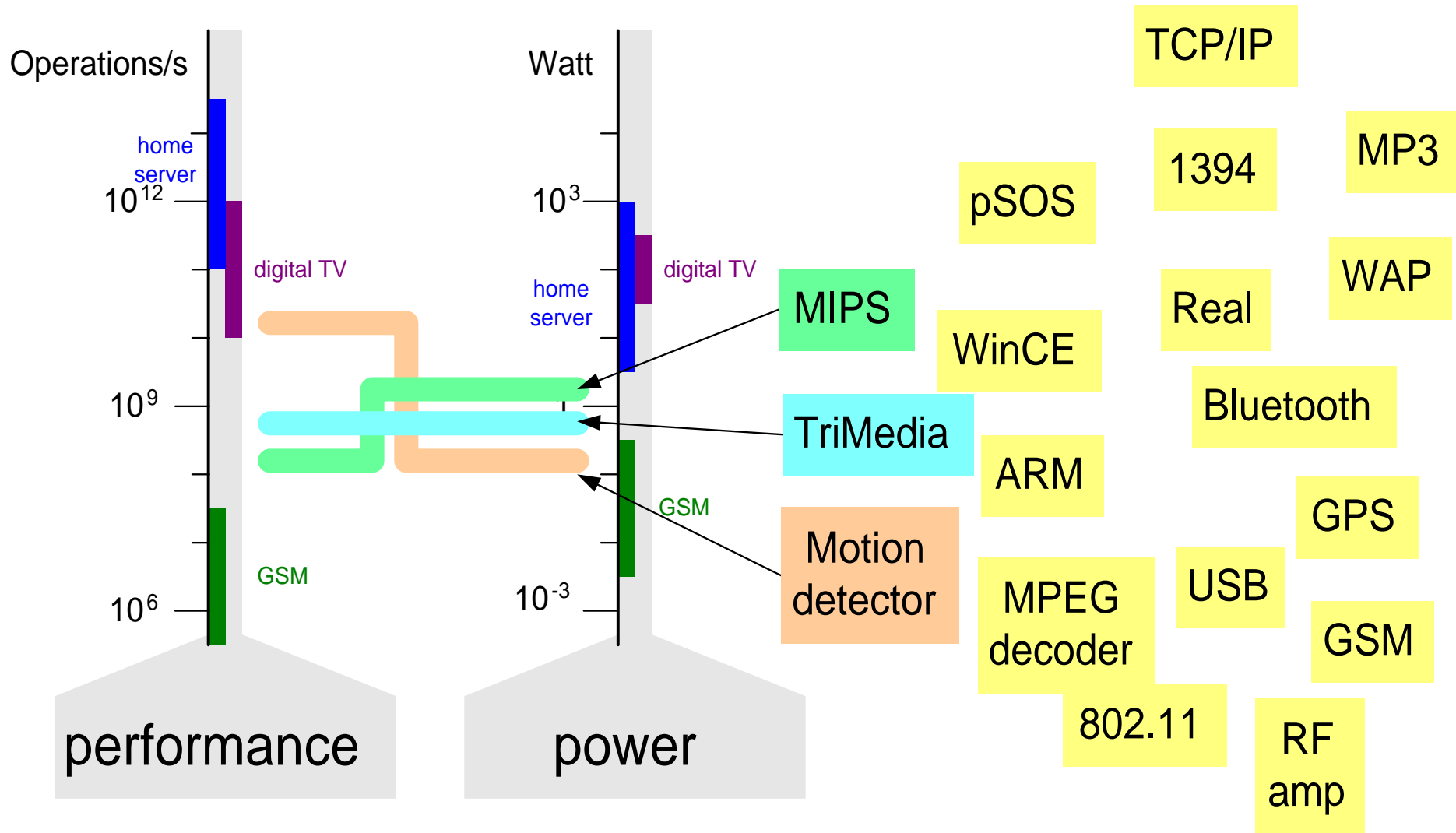
# System Integrator Problem Space - Technology



# System profile



# Semiconductors Technology solutions



# Partial Solution: Configurable Component Platform

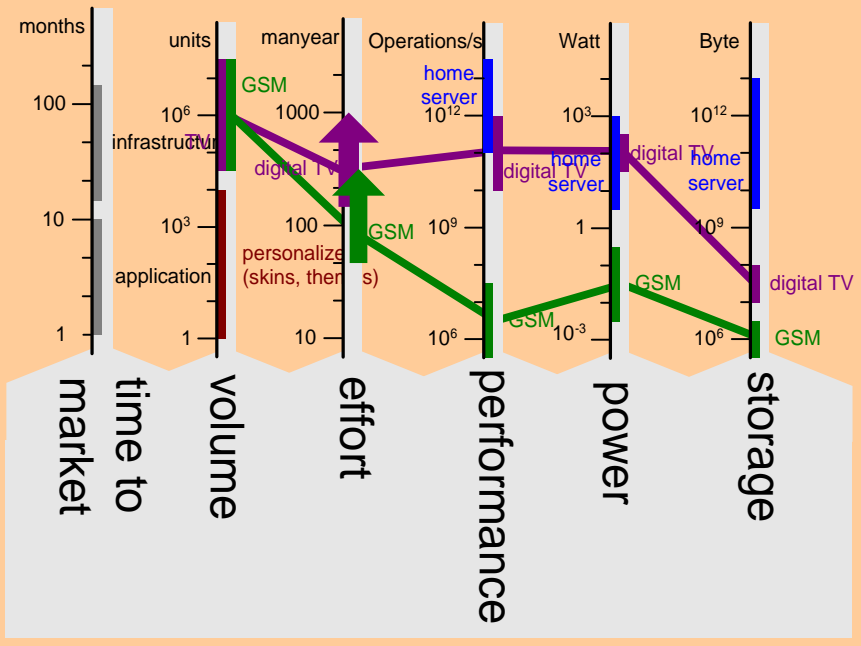
Technologies															
Systems	MIPS	TriMedia	MPEG decoder	ARM	Real	GSM	RF amp	Bluetooth	TCP/IP	MP3	pSOS	WinCE	1394	GPS	
watch				●	○	○	○	●	○	○	●	○		○	
communicator	○	○	○	●	●	●	●	○	●	○	●	○		○	
digital TV	●	●	●					○	○	○	●	○	●		
set top box	●	●	●					○	●	○	●	○	●		
pda	○	○	○	●	○	○	○	○	●	○		●		○	
camcorder	●	●	●			○	○	○	○	○	●		●	○	

● required

○ optional

# Exploring problem space and solution ingredients

## Problem space



Systems	Technologies													
	MIPS	TriMedia	MPEG decoder	ARM	Real	GSM	RF amp	Bluetooth	TCP/IP	MP3	pSOS	WinCE	1394	GPS
watch				●	○	○	○	●	○	○	●	○		○
communicator	○	○	○	●	●	●	●	○	●	○	●	○		○
digital TV	●	●	●					○	○	○	●	○	●	
set top box	●	●	●					○	●	○	●	○	●	
pda	○	○	○	●	○	○	○	○	●	○		●		○
camcorder	●	●	●			○	○	○	○	○	●		●	○

Composable Architecture

- required
- optional

Family of products

Solution ingredients

Configurability

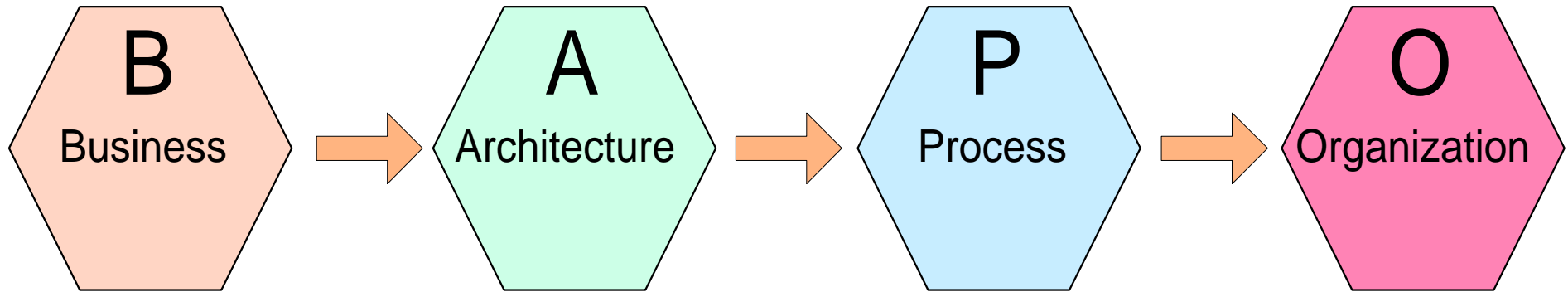
Programmability, flexibility

Increase supplier content

Competitive Performance / cost / power

# More than Architecture

---

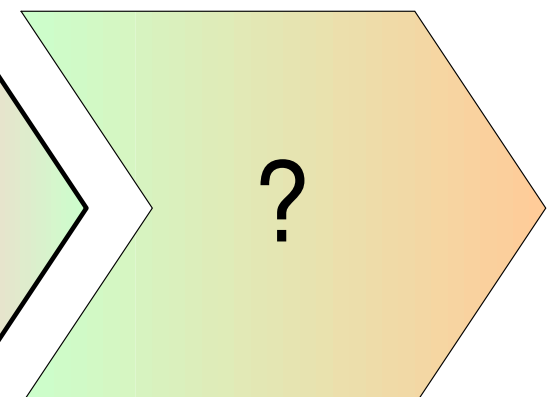
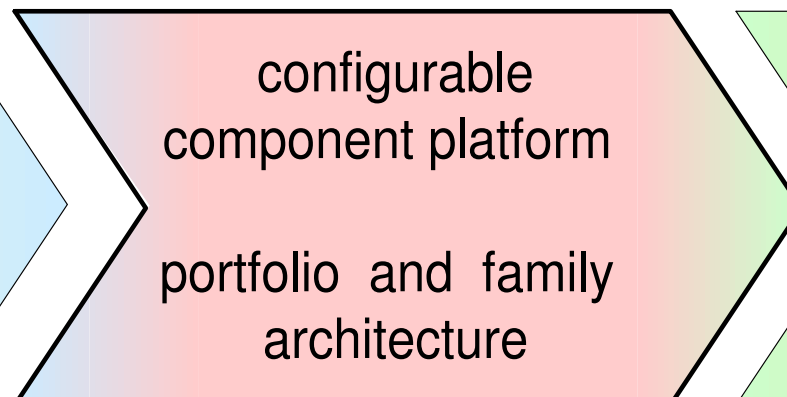
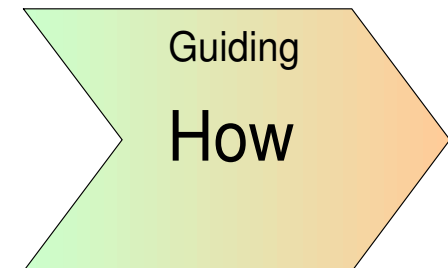
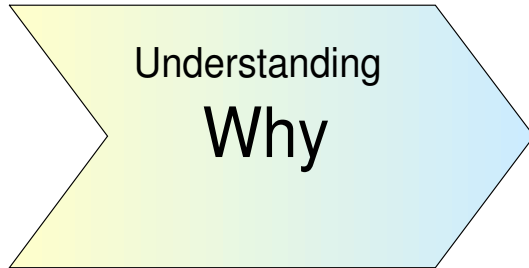


From: COPA tutorial;  
Philips SW conference 2001.

Architecture only works if the complementary viewpoints are addressed consistently

# Conclusions Part 1

---



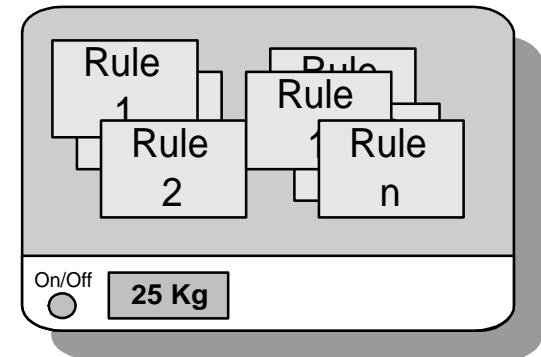
---

## Part 2:

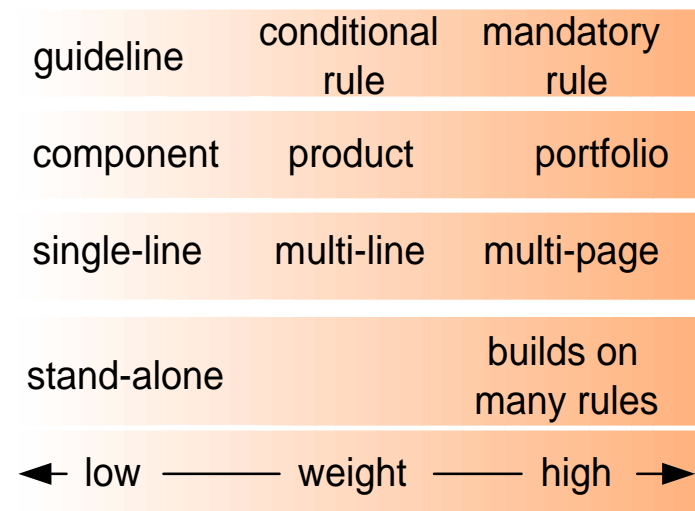
Do the things right: light-weight architecture;  
Architectural Chaos or Bureaucratic Control?

# Architecture Weight

$$\text{weight}(\text{architecture}) = \sum_{\text{all rules}} \text{weight}(\text{rule})$$



$\text{weight}(\text{rule}) = f ($ 
 level of **enforcement** ,  
**scope** (impact) ,  
**size** ,  
 level of **coupling** or  
 number of dependencies )



# Scope and Impact

---

Business or Portfolio

| n  
| m

Product Family

| n  
| m

Product

| n  
| m

Subsystem

| n  
| m

Component

Heavy-weight

High impact

Large scope

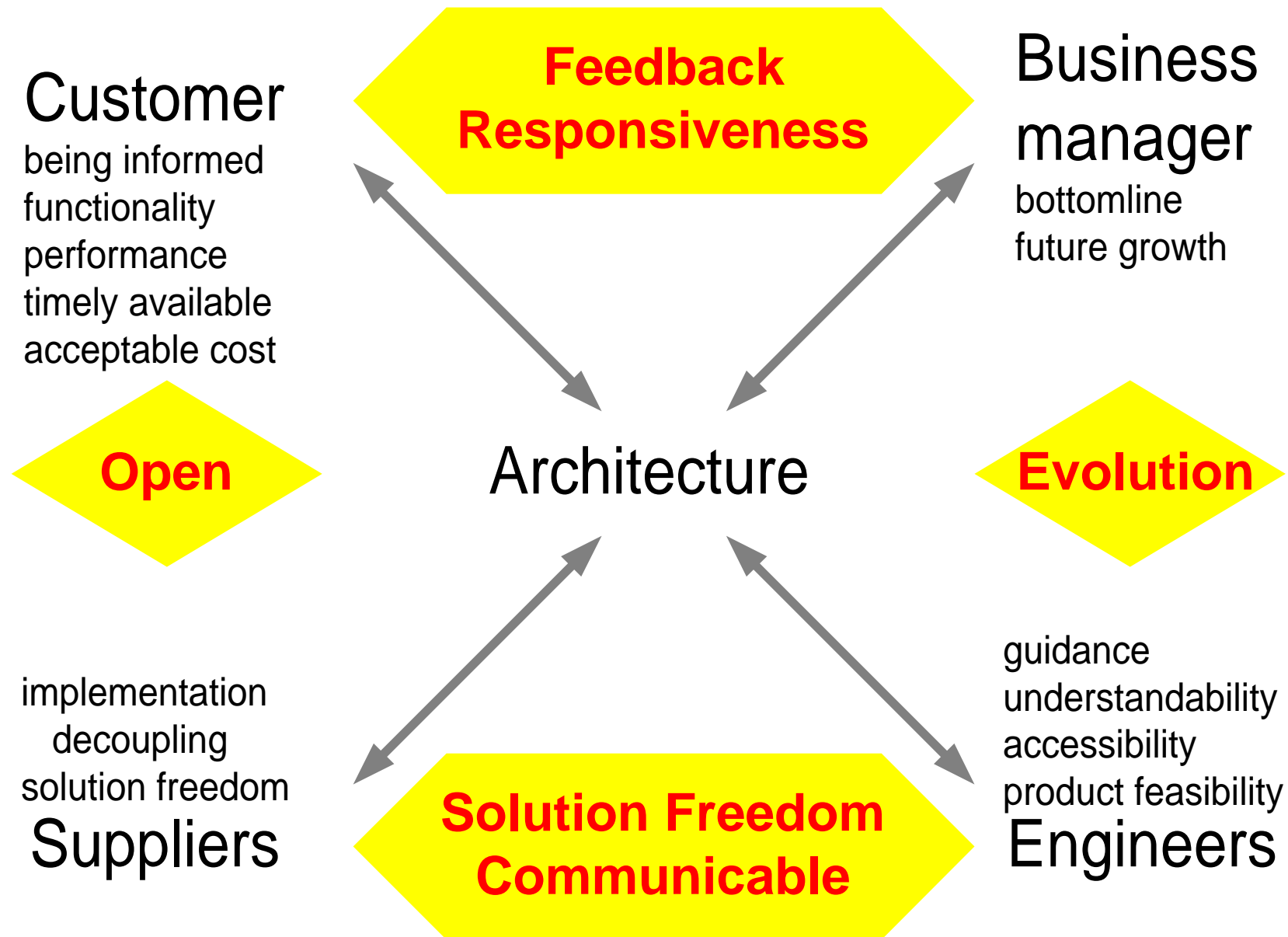


Small scope

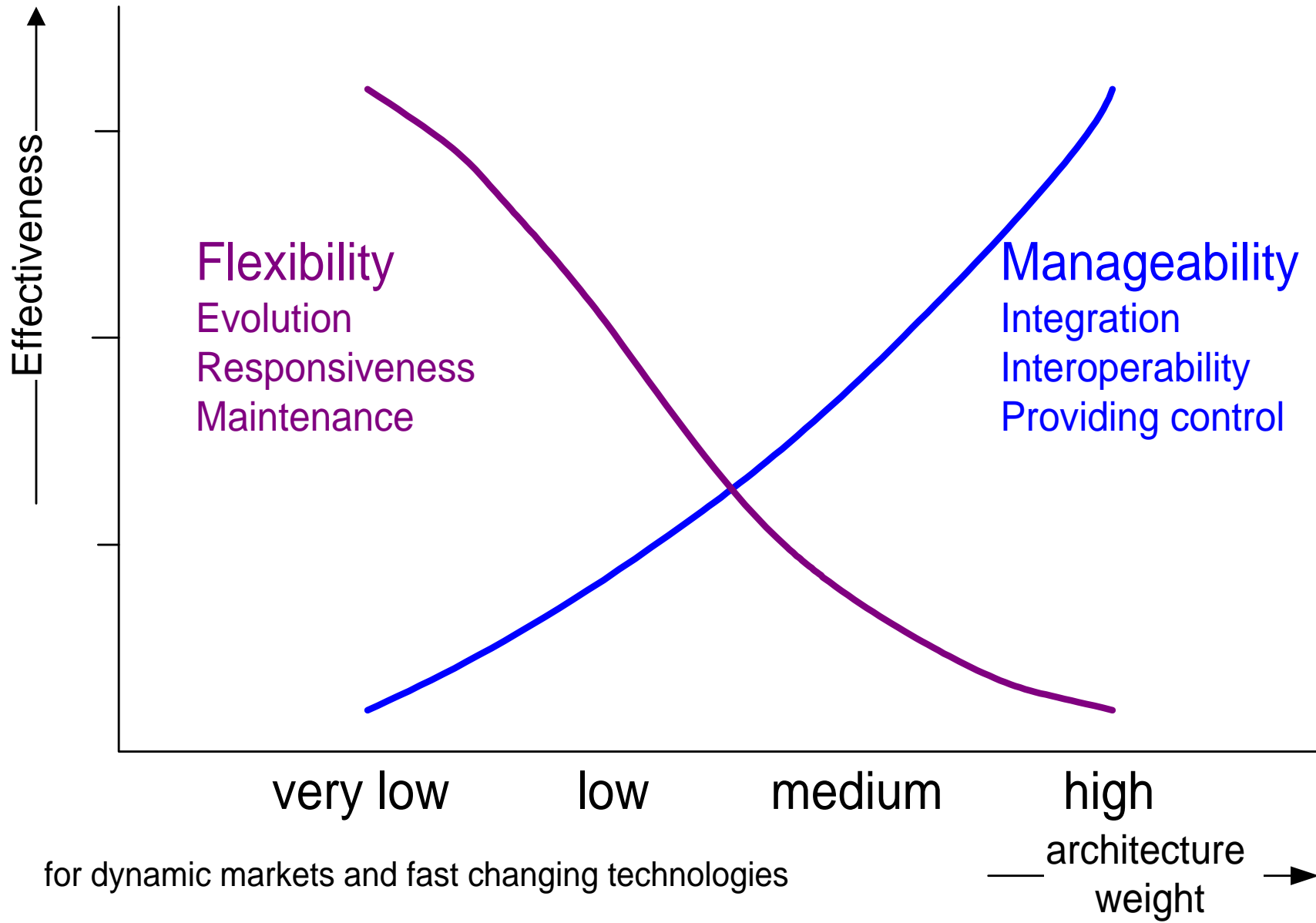
Low impact

Light-weight

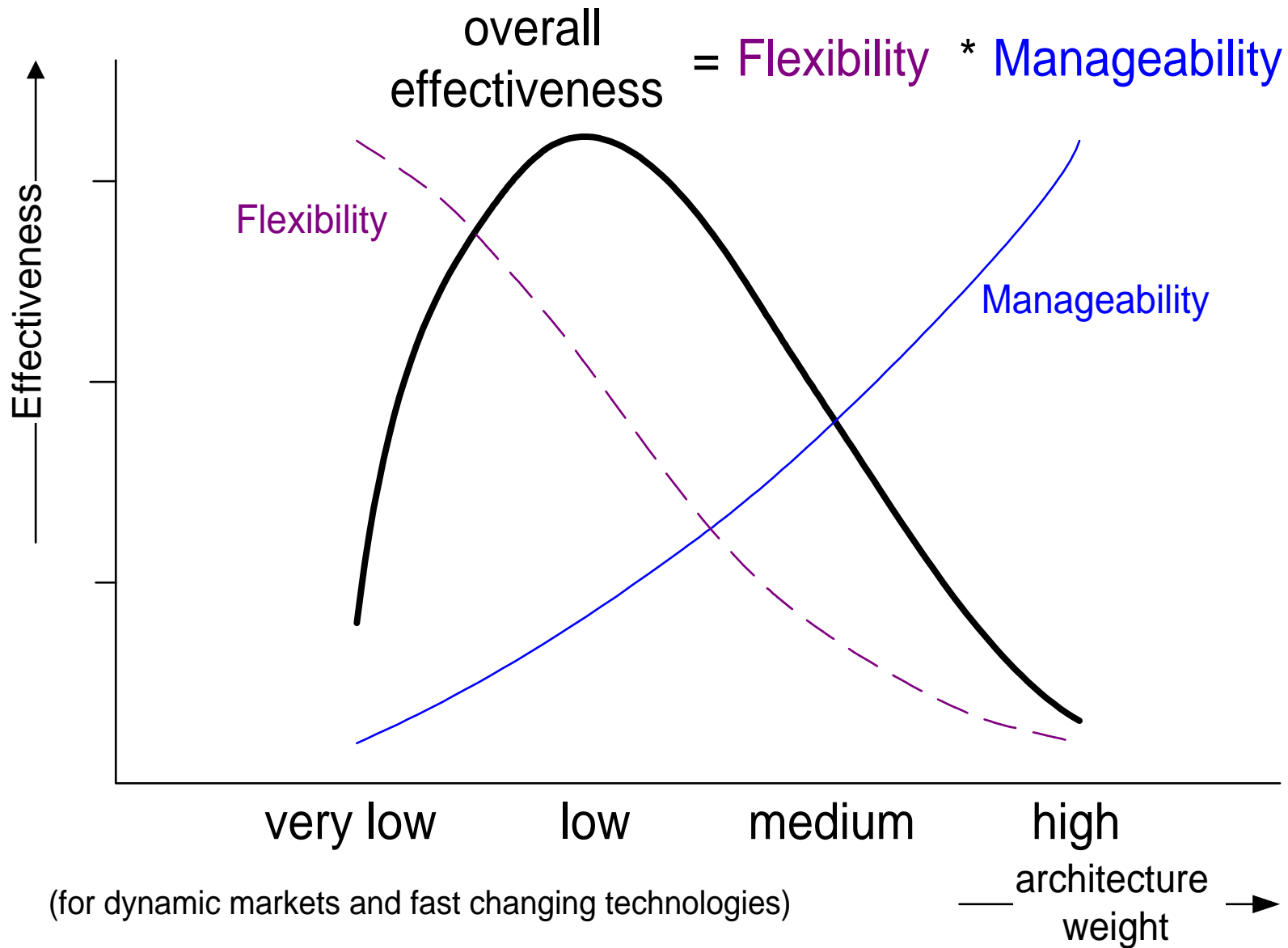
# Criteria for an Architecture



# Weight versus Effectiveness



# Conclusion Part 2



# Light Weight How -To

---

$$\text{weight}(\text{architecture}) = \sum_{\text{all rules}} \text{weight}(\text{rule})$$

2. Minimize the weight per rule

1. Reduce the rule set to the (business) essential

Understand  
your customer  
your customer's customer  
etcetera

# Minimize Rule Weight

weight(rule)=

f ( level of **enforcement** ,

**scope** (impact) ,

**size** ,

level of **coupling** or  
number of dependencies )

minimize number of mandatory rules

empower, delegate

minimize implementation details  
focus on essential concepts

Apply design principles on architecture

Multi-view architecting

# Size example: from detail to concept

Every processor will be:

minimize implementation details  
focus on essential concepts

ARM 922T

ARM 9

ARM  
ISA compatible

able to run VxWorks  
OS compatible

no constraint

heavy

medium

light

by the way, is this rule essential?

1. Reduce the rule set to the  
(business) essential

# Scope example

---

**scope** (impact) , empower, delegate

use ARM ISA compatible processor for:



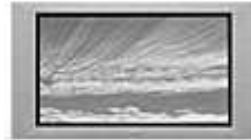
# Example product scope of rules

ARM !



GSM phone

ARM ?



flat display

ARM ?



car

ARM ??



pen



Communicator



television



watch



pda

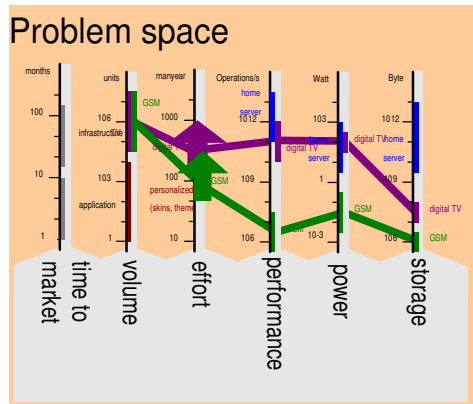
and what about OS:

PalmOS, Symbian, WindowsCE, Linux,  
VxWorks, dedicated kernel, ...

and what about programming language, storage,  
network, power, protocols, formats, user interface, ...

# Summary

## 1. Dynamic Market: Understand Your Customer



## 2. Optimal architecture:

Light weight !

