Systems on Chips

Personal computers or correct performance?

Kees Goossens
Philips Research

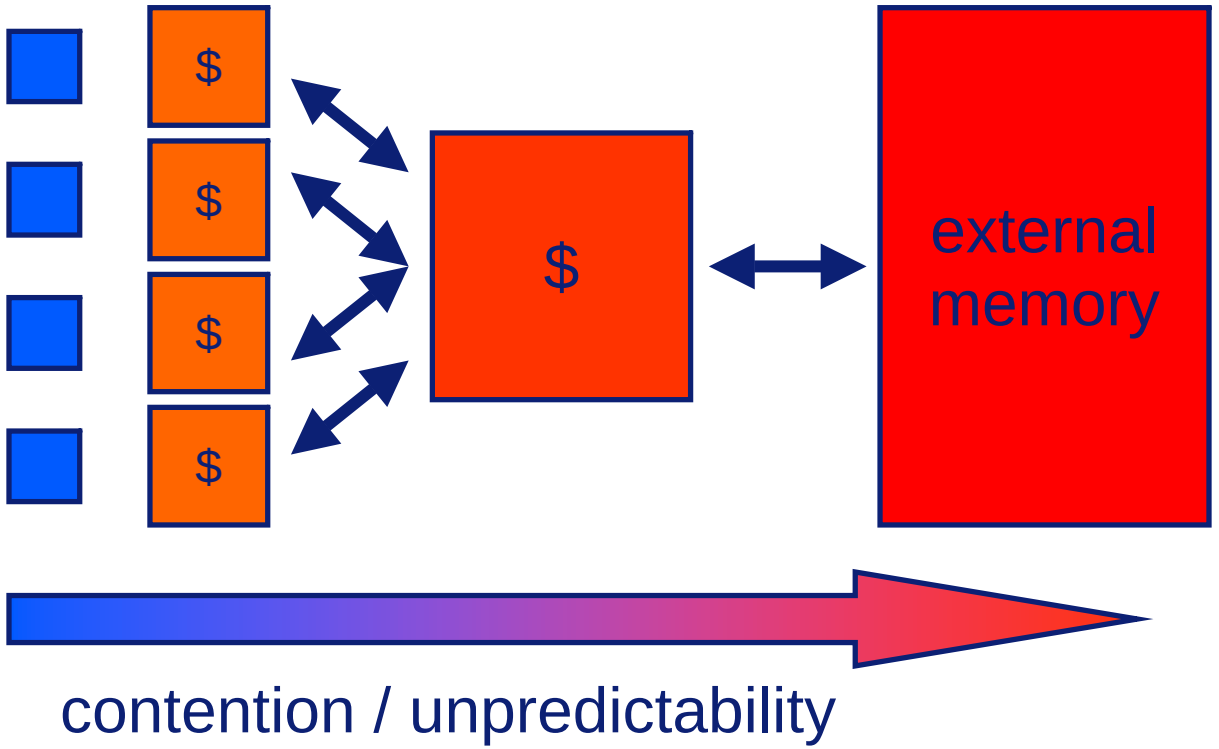
trends

- heterogeneous computation
  - processing speed increases
- communication
  - latency increases
  - throughput increases
- storage
  - memory capacity (speed) doesn't increase
  - number of external memories (1) doesn't increase

CE as a PC (personal computer)?

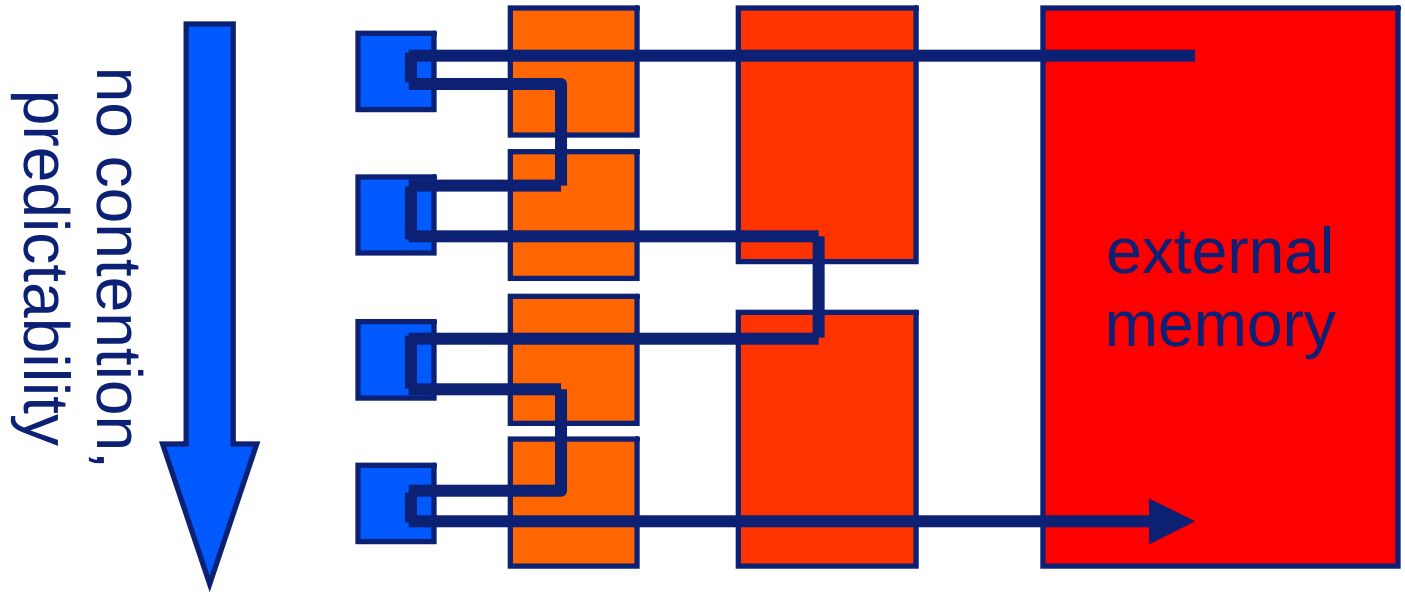
- borrow from general-purpose computing domain
- average case
- caches
- coherency

- pay *too much* to support an *inappropriate* programming model?



CE as PC (performance correct)?

- robust real-time performance
- no caches, no coherency?
- set up (virtual) pipelines
  - local computation, storage, communication

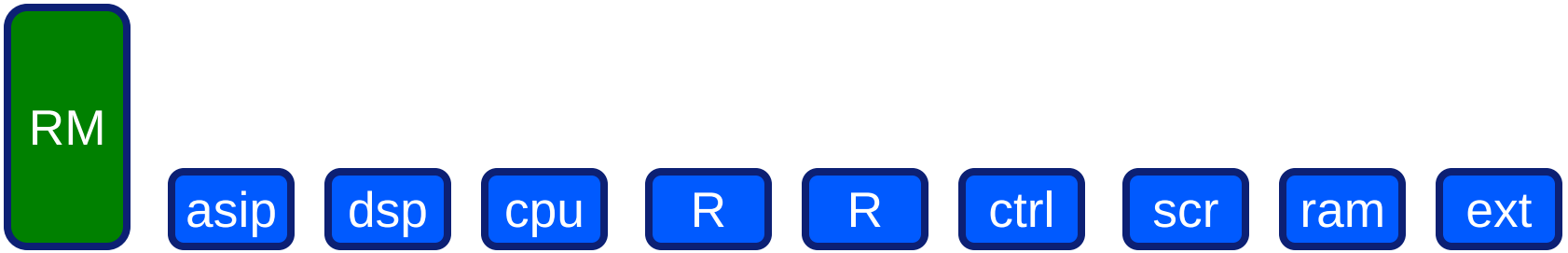


ingredients

- resource management
  - manage raw / gross resource capacity
- to offer guaranteed services
  - guaranteed nett / user resource capacity
- together gives virtualisation
  - virtual wire, virtual cpu, memory virtualisation
  - isolation, robustness, sandboxing / legacy, management
- quality of service & application manager
  - mapping of virtual pipelines / task graphs to service providers (resources)

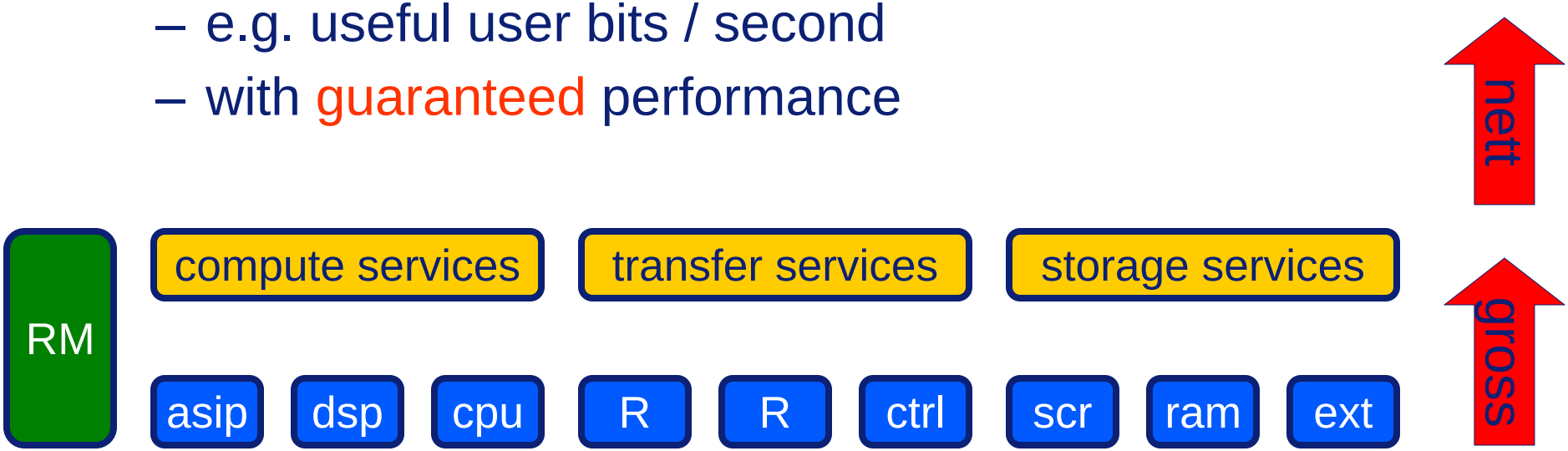
See also: Kees Goossens, et al. **Service-based design of systems on chip and networks on chip.** In Peter van der Stok, editor, Dynamic and Robust Streaming Between Connected Consumer-Electronics Devices. Springer, 2005.

service-based system design



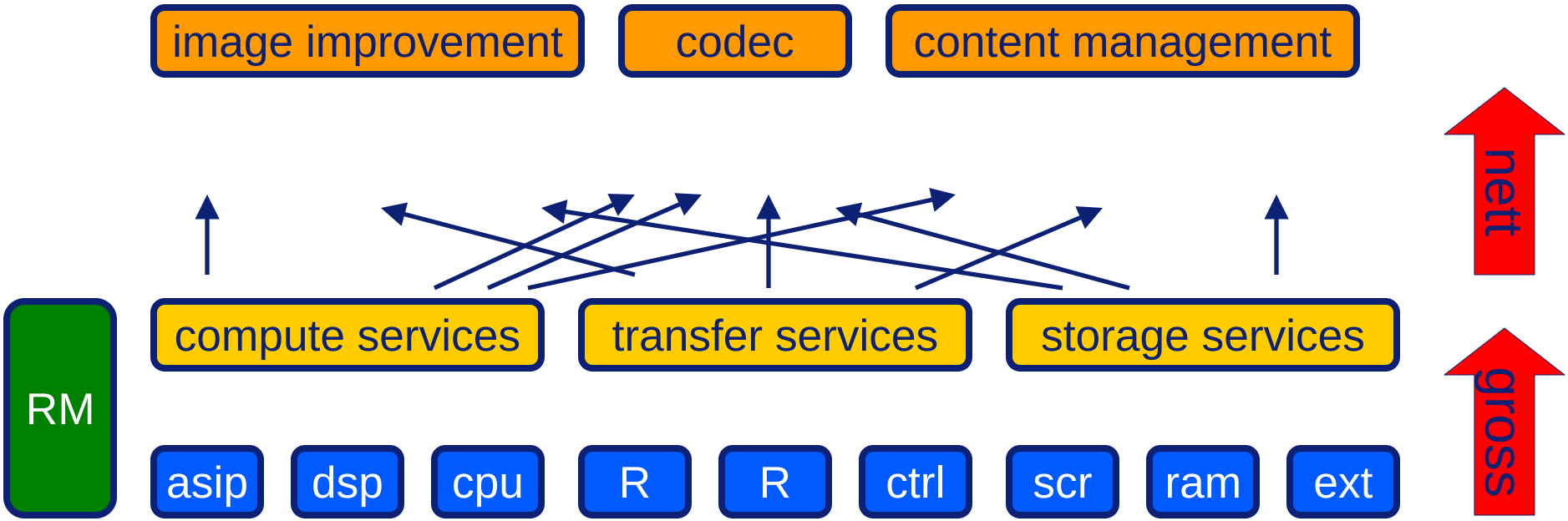
service-based system design

- resources offer gross services
  - e.g. bandwidth in terms of cycles / second
- resource manager offers nett services
  - e.g. useful user bits / second
  - with guaranteed performance



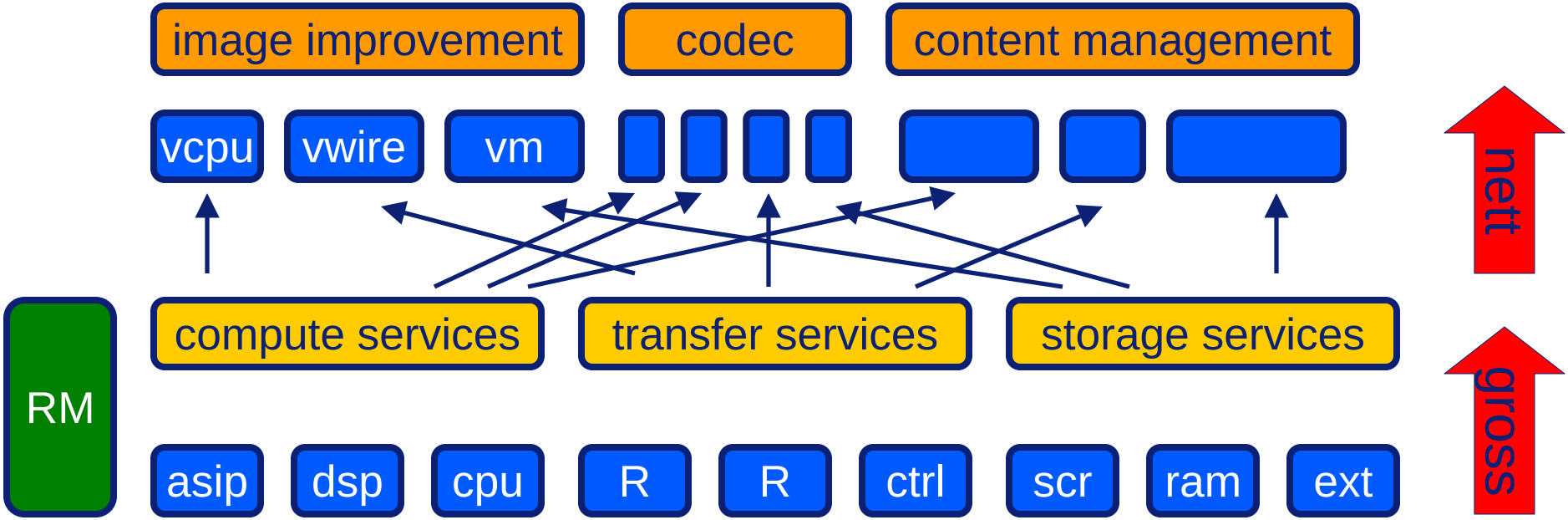
service-based system design

- compose services to create higher-level services

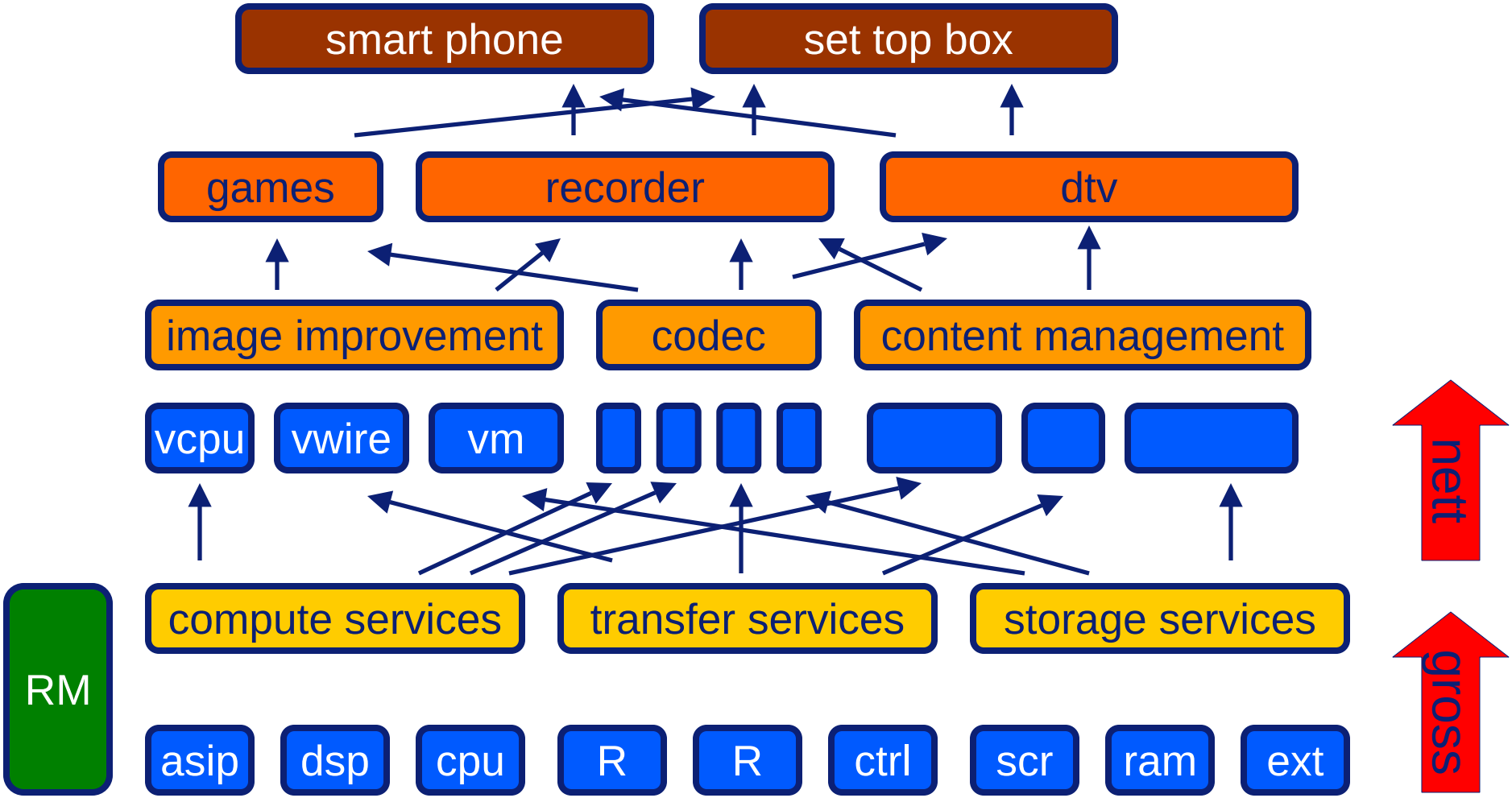


service-based system design

- guaranteed services offer
  - virtualisation of resources & services
  - compositional (RT) performance

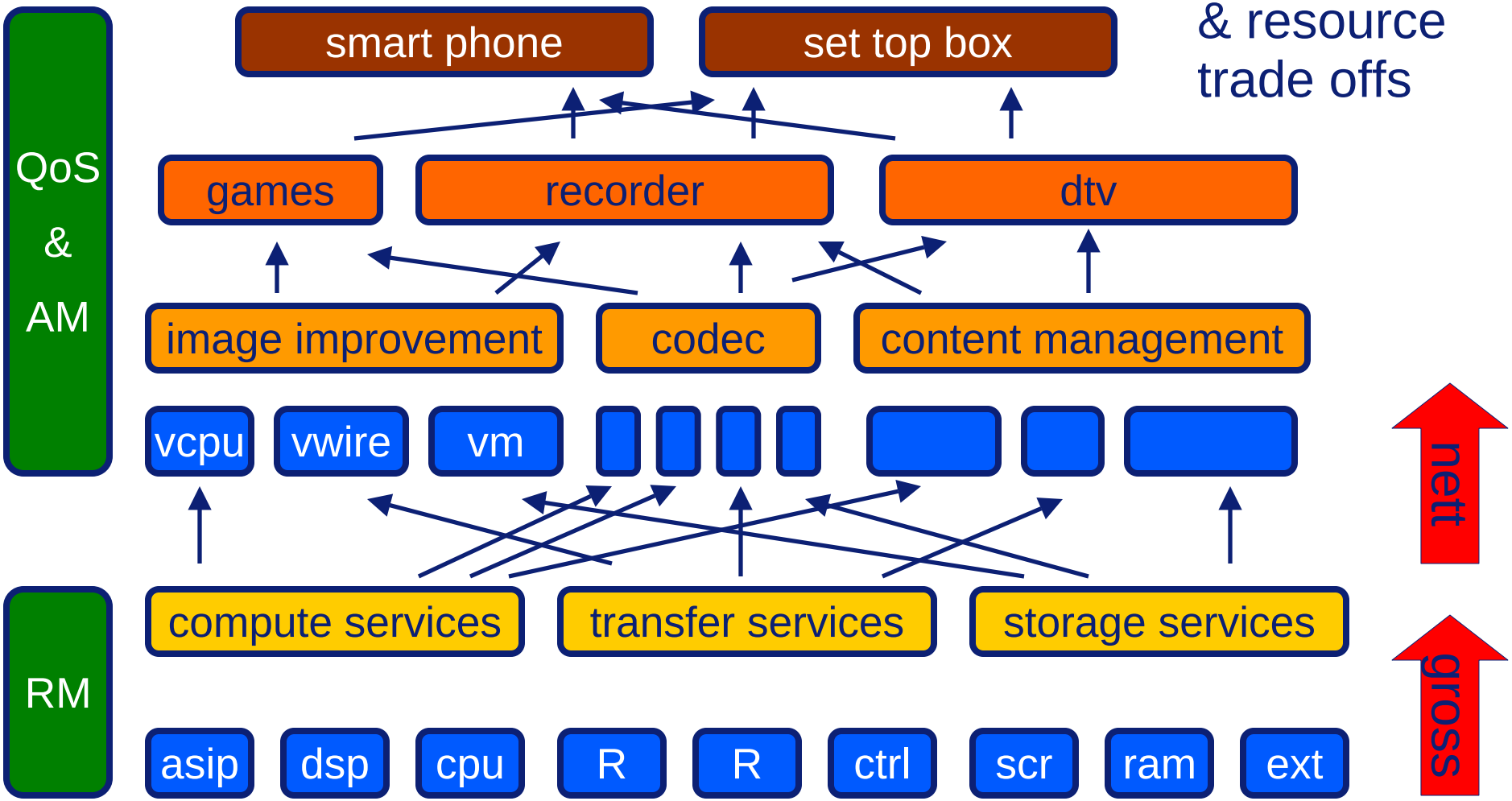


service-based system design

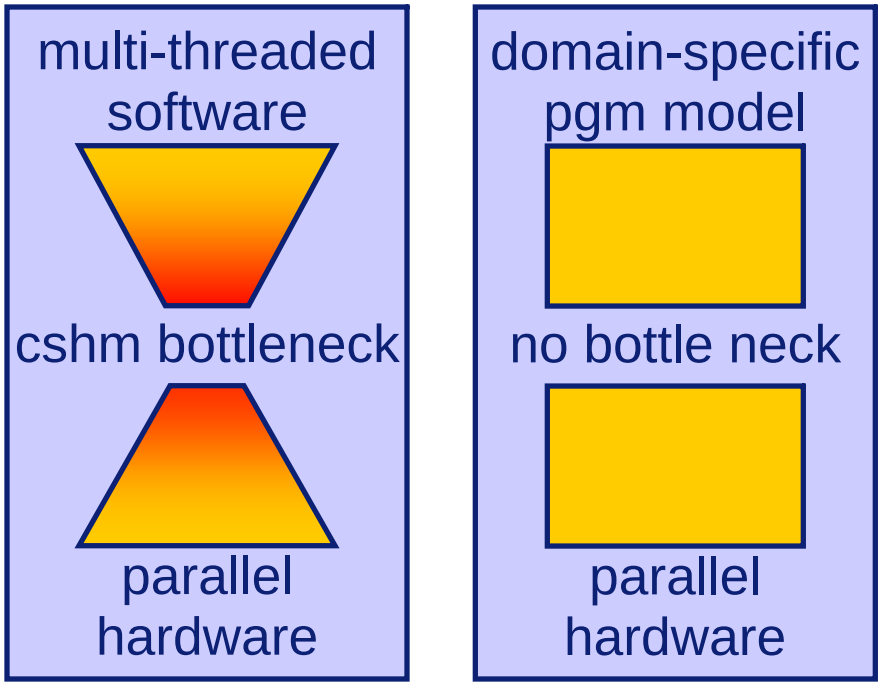


service-based system design

- QoS / AM ensures global service & resource trade offs



service abstraction



- “bare iron”
- device level
- resource level
- RTOS
- programmer’s view
  - ((coherent) shared) memory abstraction
  - domain-specific
    - (rich) components
    - streaming
    - graphics / gaming

status

- communication / networks on chip
  - QoS is hot topic
- computation
  - renaissance of virtualisation
- communication
  - external memory
  - service-based view on memory organisation


PHILIPS