Modeling and Analysis: System Model

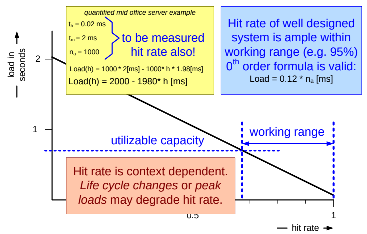
by *Gerrit Muller* University of South-Eastern Norway-NISE
e-mail: gaudisite@gmail.com
www.gaudisite.nl

Abstract

This presentation uses a web shop service as example system to construct a system model. The caching of pictures of the products in the shop is modeled to analyze performance, robustness, scalability and reliability of the system.

Distribution

August 21, 2020
status: preliminary draft
version: 0.4



Content of Modeling and Analysis System Model

content

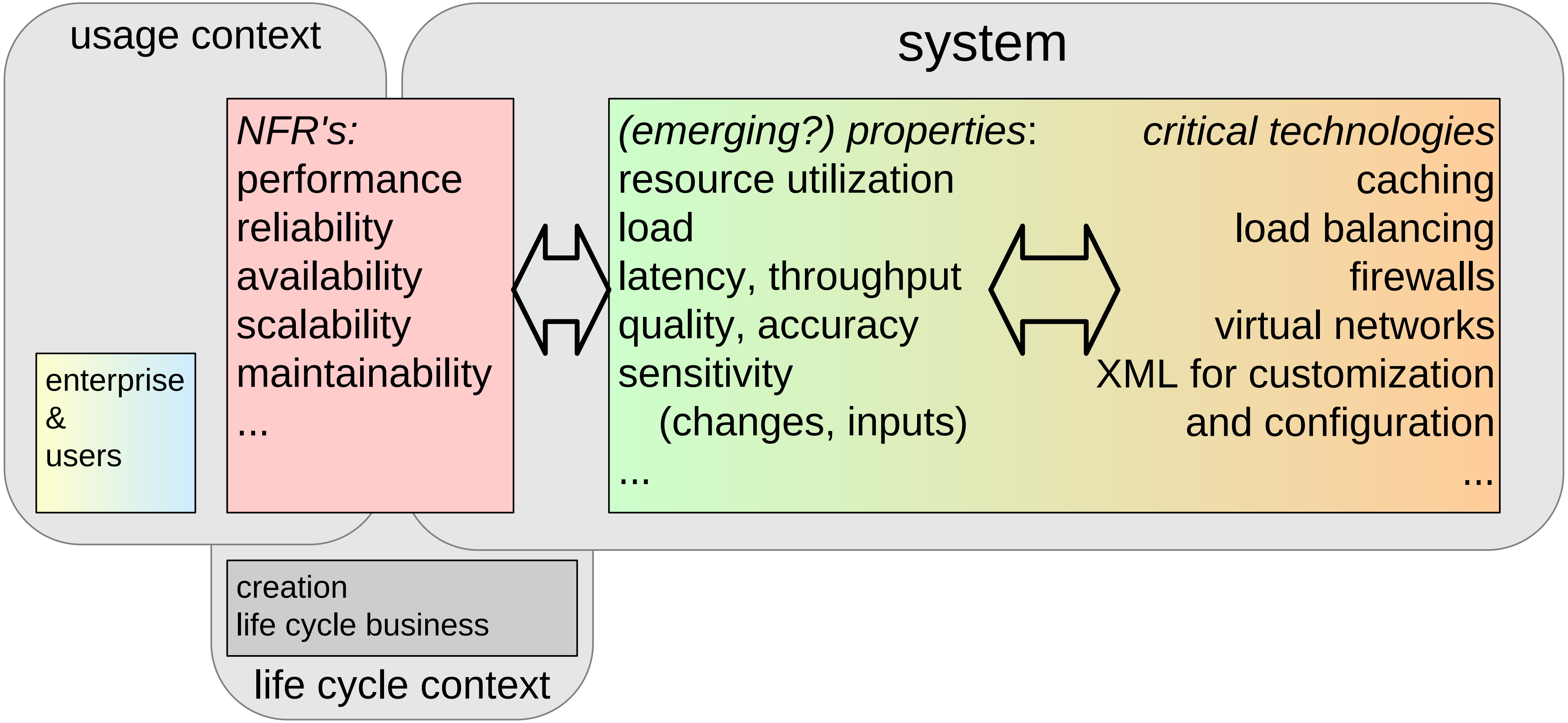
What to model of the system

Stepwise approach to system modeling

Non Functional requirements (NFR), System Properties and Critical Technologies

Examples of web shop case

What to Model in System Context?



Approach to System Modeling

1. determine relevant Non Functional Requirements (NFR's)
2. determine relevant system design properties
3. determine critical technologies
4. relate NFR's to properties to critical technologies
5. rank the relations in relevancy and criticality
6. model relations with a high score

Web Shop: NFR's, Properties and Critical Technologies

system

1

NFR's:
performance browsing
initial cost
running costs
reliability/availability
scalability order rate
maintainability
- effort product changes
- effort staff changes

security

2

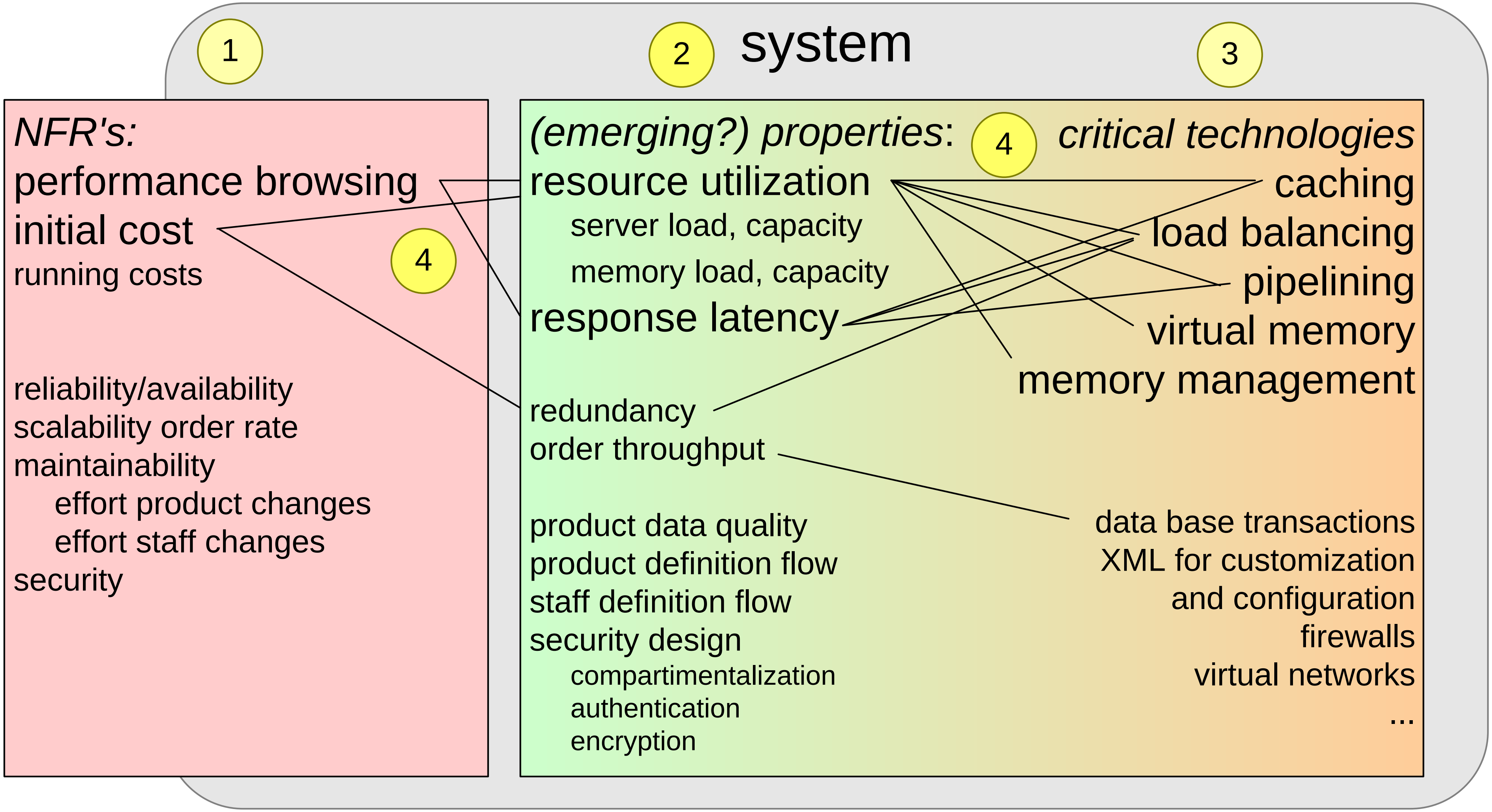
(emerging?) properties:
resource utilization
- server load, capacity
- memory load, capacity

response latency
redundancy
order throughput
product data quality
product definition flow
staff definition flow
security design
- compartimentalization
- authentication
- encryption

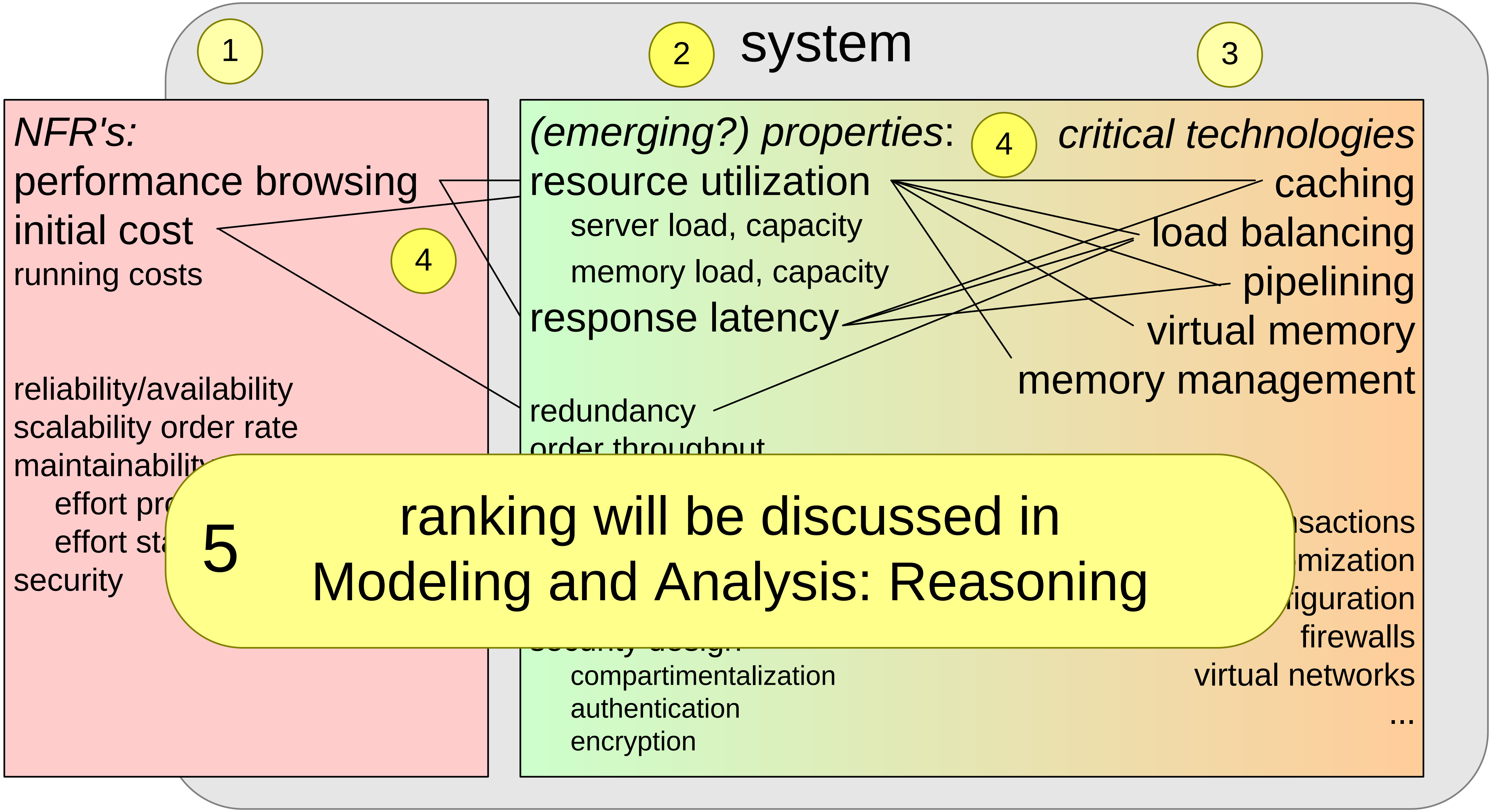
3

critical technologies
caching
load balancing
pipelining
virtual memory
memory management
data base transactions
XML for customization
and configuration
firewalls
virtual networks
...

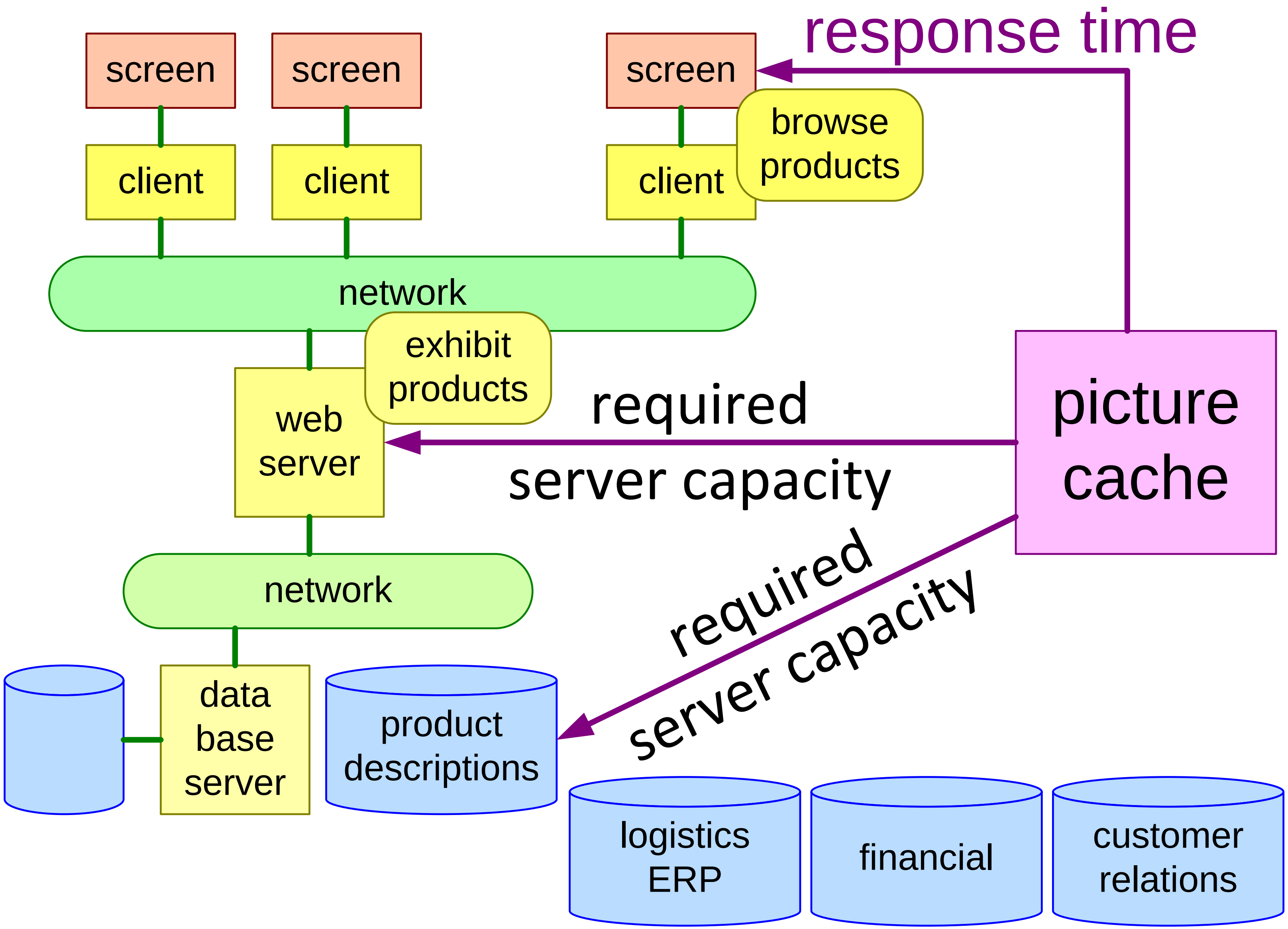
4. Determine Relations



5. Rank Relations



Purpose of Picture Cache Model in Web Shop Context



Zero Order Load Model

zero order web server load model

$$Load = n_a * t_a$$

n_a = total requests

t_a = cost per request

First Order Load Model

first order web server load model

$$Load = n_{a,h} * t_h + n_{a,m} * t_m$$

$n_{a,h}$ = accesses with cache hit

$n_{a,m}$ = accesses with cache miss

t_h = cost of cache hit

t_m = cost of cache miss

$$n_{a,h} = n_a * h$$

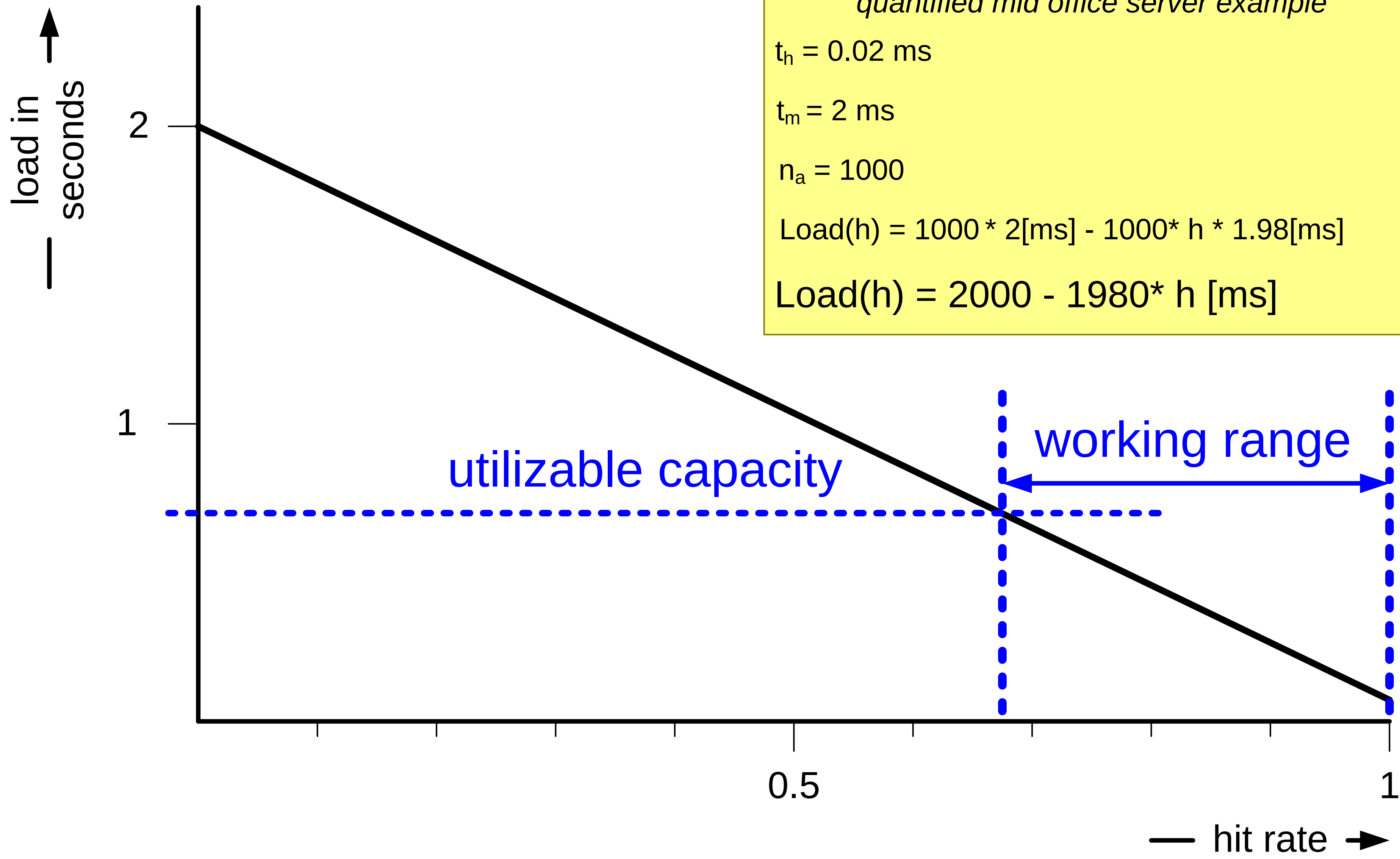
$$n_{a,m} = n_a * (1-h)$$

n_a = total accesses

h = hit rate

$$Load(h) = n_a * h * t_h + n_a * (1-h) * t_m \qquad = n_a * t_m - n_a * h * (t_m - t_h)$$

Quantification: From Formulas to Insight



quantified mid office server example

t_h = 0.02 ms

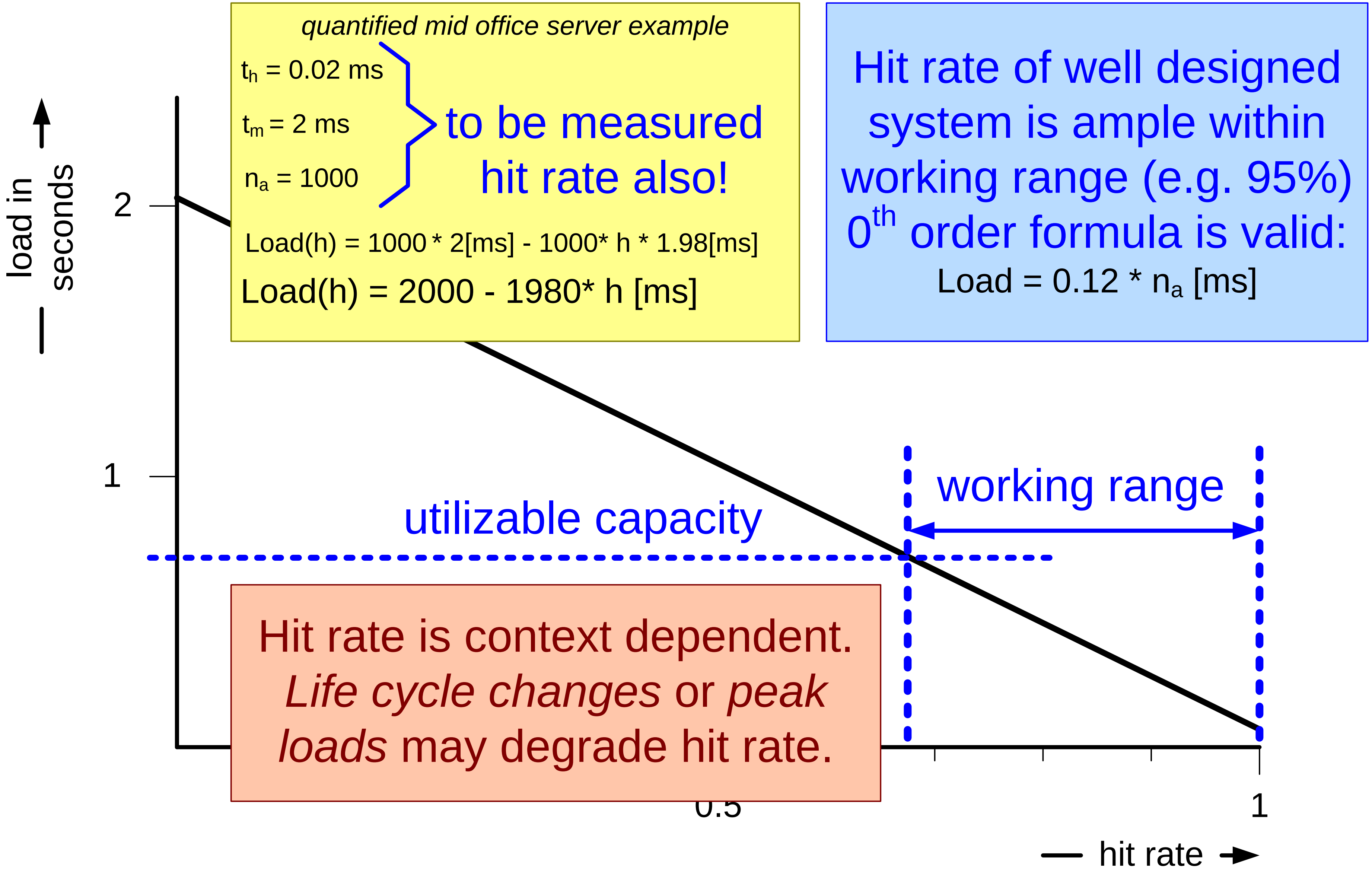
t_m = 2 ms

n_a = 1000

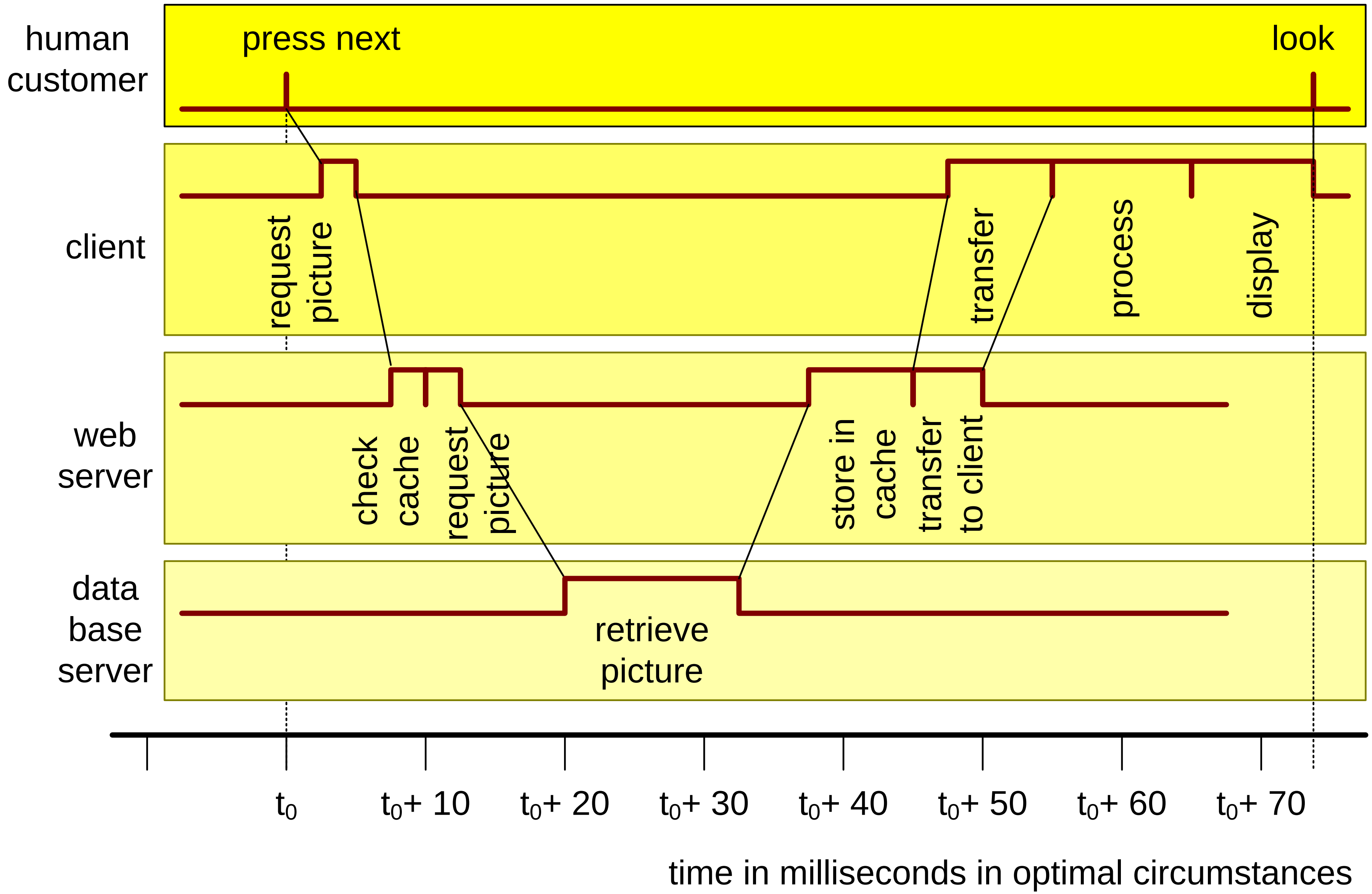
Load(h) = 1000 * 2[ms] - 1000* h * 1.98[ms]

Load(h) = 2000 - 1980* h [ms]

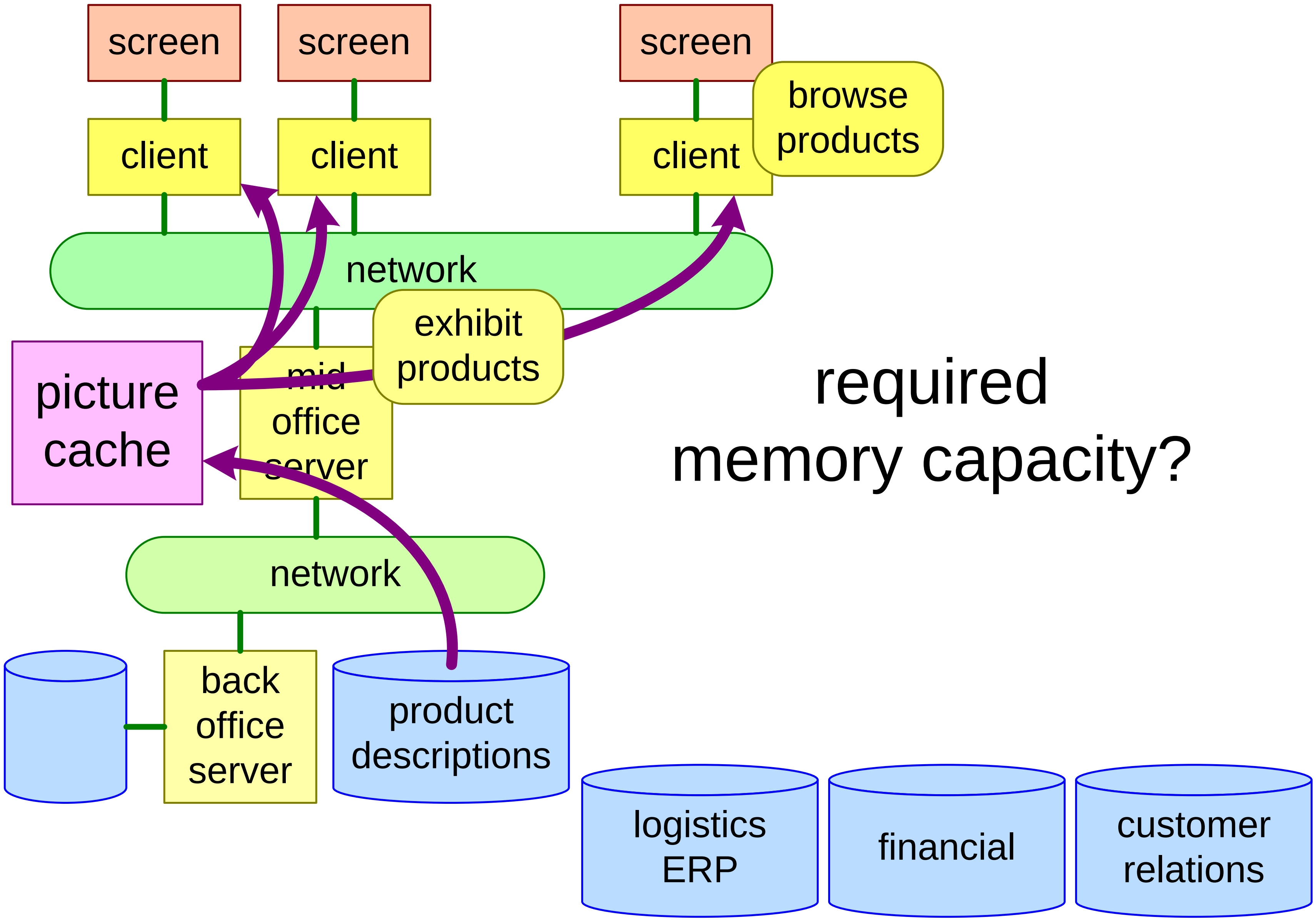
Hit Rate Considerations



Response Time

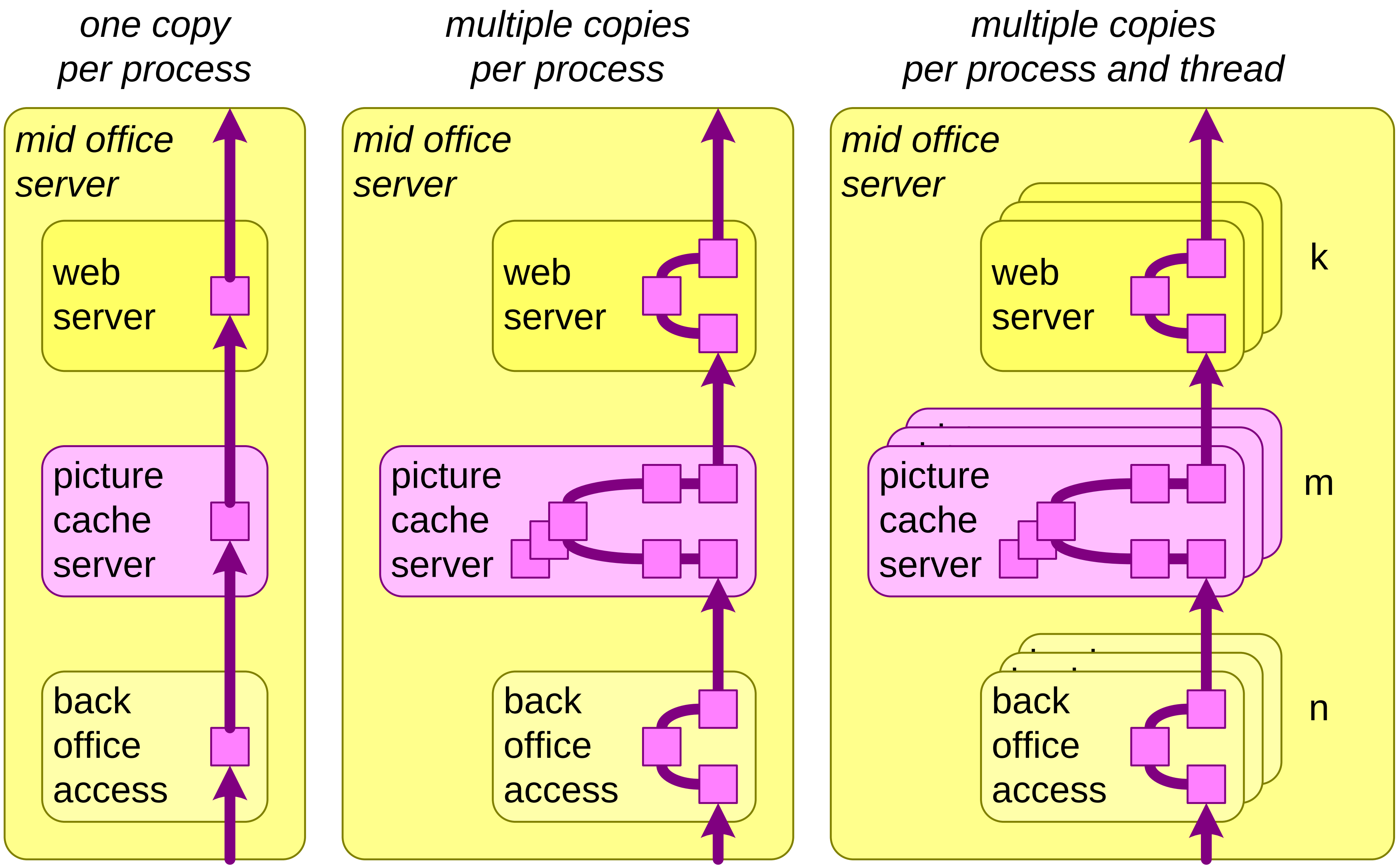


What Memory Capacity is Required for Picture Transfers?



required memory capacity?

Process View of Picture Flow in Web Server



Formula memory Use Web Server

picture memory =

3 * n * s +

5 * m * s + c * s +

3 * k * s

where

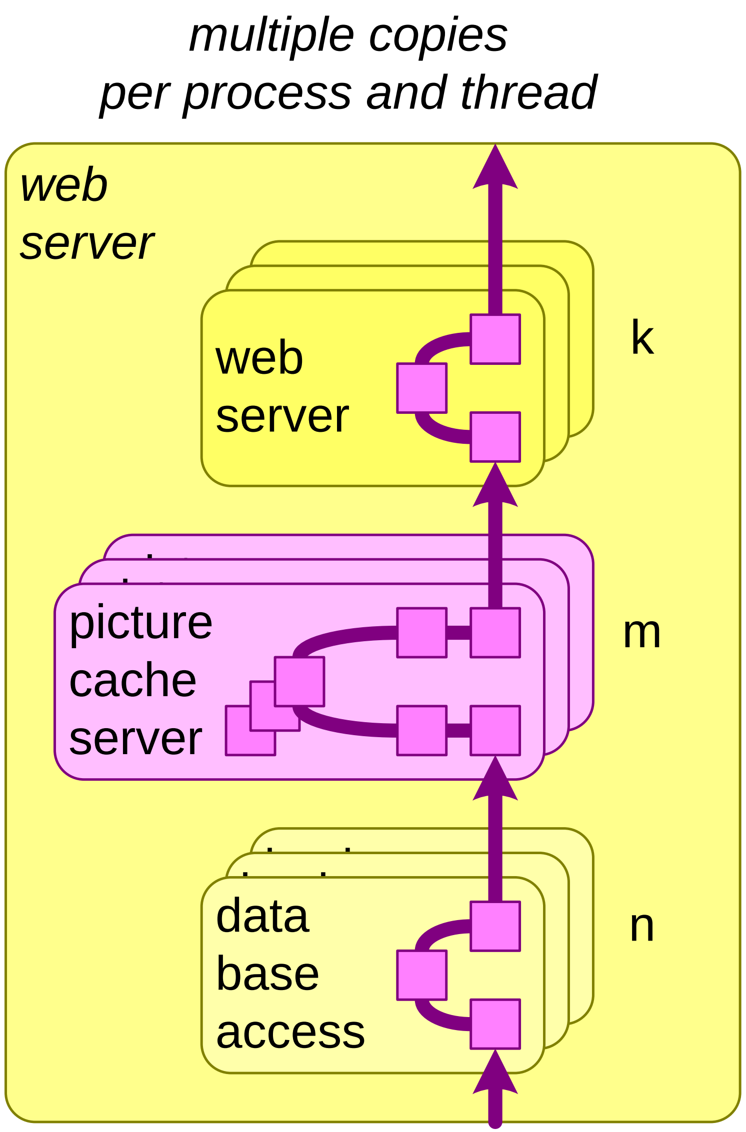
n = # data base access threads

m = # picture cache threads

k = # web server threads

s = picture size in bytes

c = in memory cache capacity in # pictures



Web Server Memory Capacity

use case	n	m	k	s	c	MB	storage type
small shop	1	1	1	100	10	1.5	L3
	2	4	10	100	20	5.3	main
highly concurrent	2	4	1000	100	100	296	main
large pictures	2	4	1000	1000	100	2,962	main+disk
many pictures	2	4	10	100	100,000	9,540	main+disk
all at once	2	4	1000	1000	100,000	98,234	disk

picture memory = 3 n s + 5 m s + c s +3 k s

where

n = # back office access threads

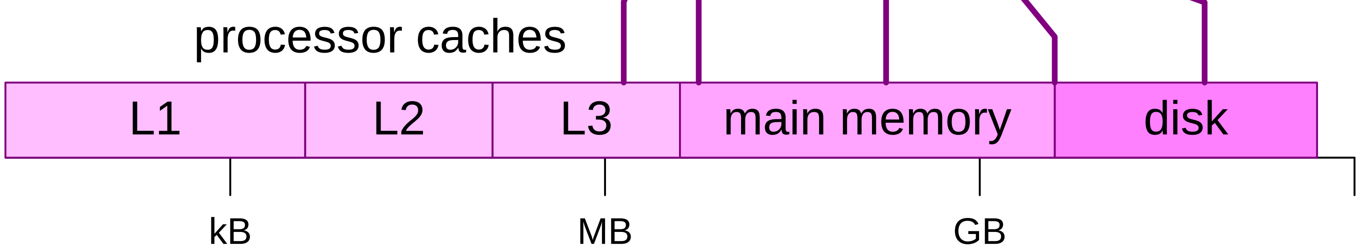
m = # picture cache threads

k = # web server threads

s = picture size in bytes

c = in memory cache capacity in # pictures

memory use
- product browsing only
- pictures only
- single server



What is the performance impact of memory use on other processing?

We Have only Modeled a Small Part of the System...

function	browse/exhibit products	sales, order intake, payments track, order handling stock handling financial bookkeeping customer relation management update catalogue advertize after sales support
data	picture	structured (product attributes, logistics, ...) program code
aspect	server memory use response time server load	network use reliability any resource, any NFR

$$\text{aspect result} = \sum_{d\,=\,\text{all data}} \; \sum_{f\,=\,\text{all functions}} \text{aspect}(d, f)$$

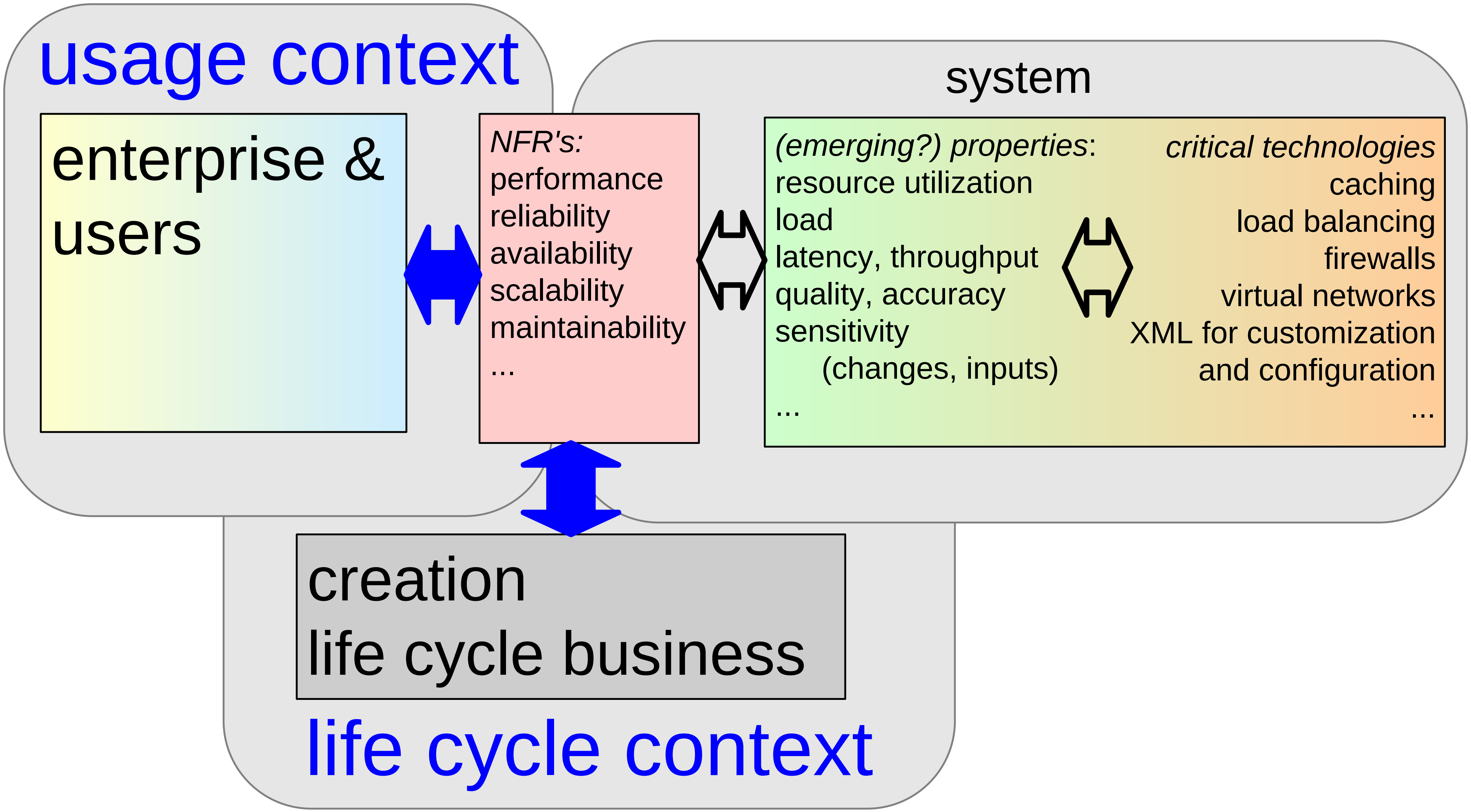
ignoring other dimensions such as applications, users, circumstances static

mostly assumptions and coarse estimates

some insight in:

- what are key design issues
- what are relevant use case areas

Refinement After Context Modeling



Summary

Conclusions

Non Functional Requirements are the starting point for system modeling

Focus on highest ranking relations between NFR's and critical technologies

Make simple mathematical models

Evaluate quantified instantiations

Techniques, Models, Heuristics of this module

Non functional requirements

System properties

Critical technologies

Graph of relations