



Software Mobiliteit

UAMS – 6 maart 2001

Theo D'Hondt

Lab voor Programmeerkunde

Vrije Universiteit Brussel

<http://prog.vub.ac.be/~tjdhondt>

Overzicht

- ☑ **Stelling**
- ☑ **Objecttechnologie**
- ☑ **Distributie**
- ☑ **Mobiliteit**
- ☑ **Evolutie**
- ☑ **Besluit**

Stelling

mobiele software

=

((((objecten) + distributie) + migratie)

Objecttechnologie: historiek

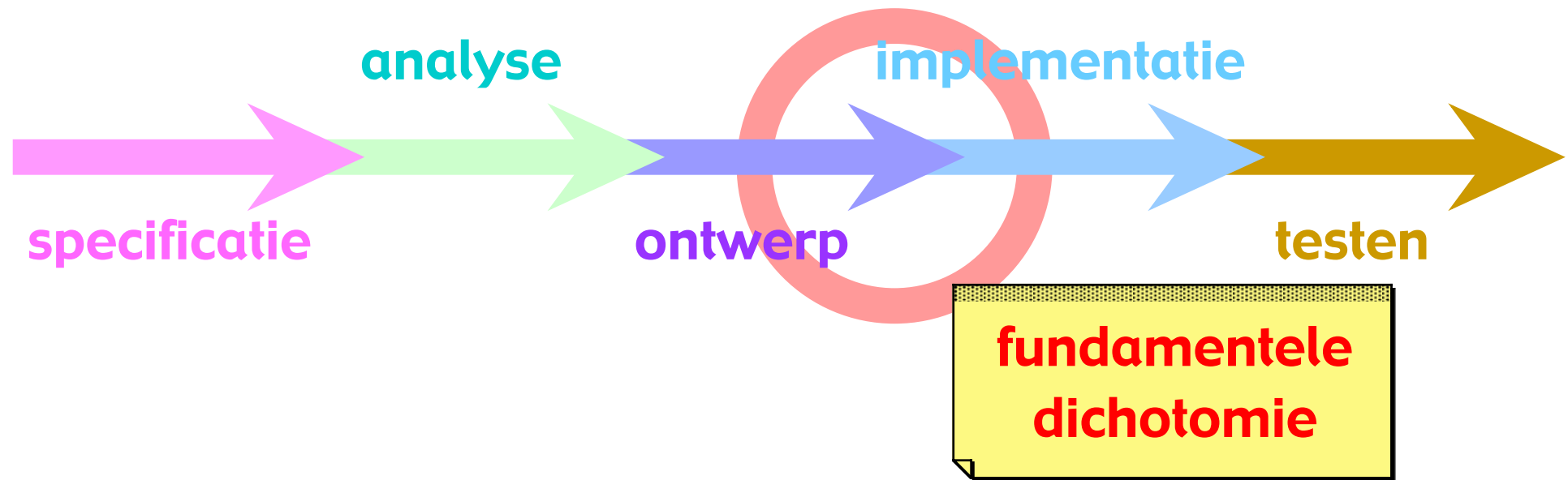
- ✓ OO begonnen met OO programmeren in '67
- ✓ OO op punt gesteld vanaf '70 met Smalltalk
- ✓ OO werd een professionele tool vanaf '80 met C++
- ✓ OO werd de focus van IT vanaf '90
- ✓ OO is vandaag universeel door internet & Java

Objecttechnologie: essentie

Beheersen van complexe software via

-  **encapsulatie** (data abstractie)
-  **incrementele** specificatie (polymorfisme)

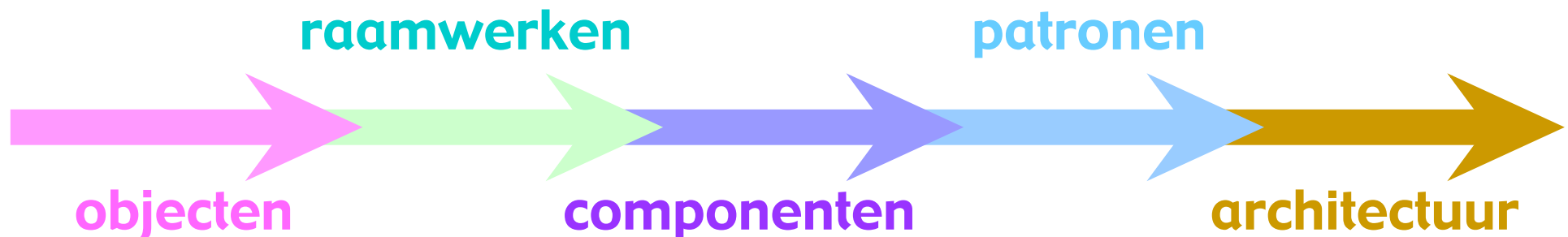
Objecttechnologie: crisis



Objecttechnologie: crisis

- ☑ **objecten zijn partiële oplossing**
- ☑ **objecten zijn niet schaalbaar**
- ☑ **objecten vragen expertise**
- ☑ **architectuur!**

Objecttechnologie: trends

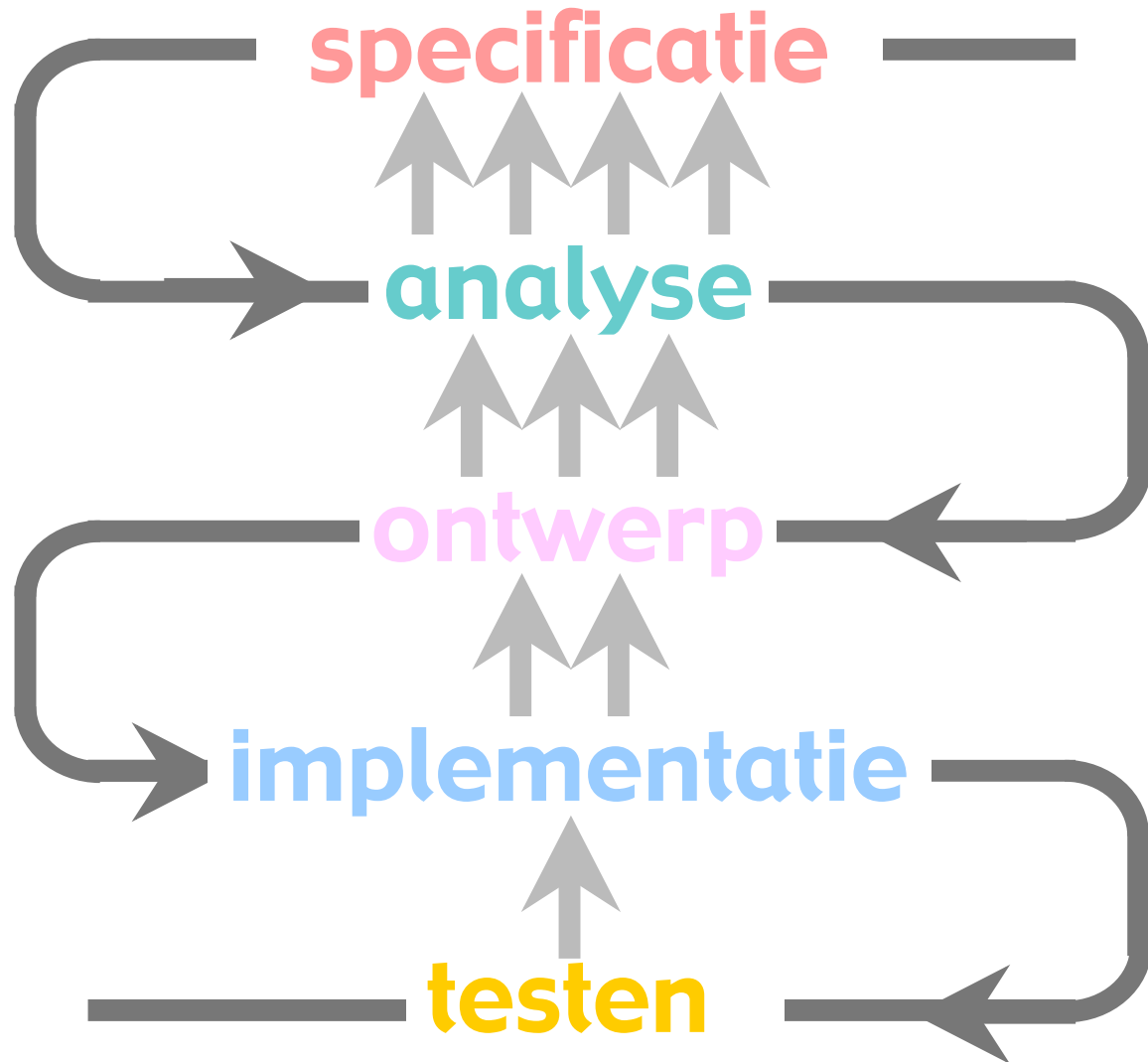


**code hergebruik \neq
conceptueel hergebruik**

Objecttechnologie: inzicht

De handeling van **implementatie** van een complex software artefact draagt op fundamentele wijze bij tot het **begrijpen** ervan

Objecttechnologie: nù



prototypen

Objecttechnologie: boutade

software

=

UML + XML + Java

concepten

"glue"

implementatie

Objecttechnologie: 2 kampen



UP



XP

*unified
process*

*extreme
programming*

**gezond
verstand?**

Objecttechnologie: *UP*

- geplande levenscyclus**
- gestuurd door model**
- roundtrip engineering***
- voorspelbaar**
- expertise?**

Objecttechnologie: *XP*

- geplande ontwikkelcyclus**
- gestuurd door implementatie**
- code documenteert alles**
- voorspelbaar**
- expertise!**

Objecttechnologie: besluit

- ☑ **dominant in softwareontwikkeling**
- ☑ **gedomineerd door Java**
- ☑ **enorme verspreiding & veralgemening**
- ☑ **overdreven enge visie**
- ☑ **historisch bewustzijn?**

Distributie: systeem

Een gedistribueerd systeem is een verzameling **meervoudige**, **interagerende** virtuele machines

een enkelvoudig systeem is een speciaal geval

communicatie of gemeenschappelijke *resources*

software of hardware machines

Distributie: applicatie

Een gedistribueerde applicatie is een verzameling **meervoudige**, **interagerende** processen

van eenvoudige *task*
tot uitbatingssysteem

versturen van berichten
of delen van geheugen

vertolkt door
programma of
computer

Distributie: pijnpunten

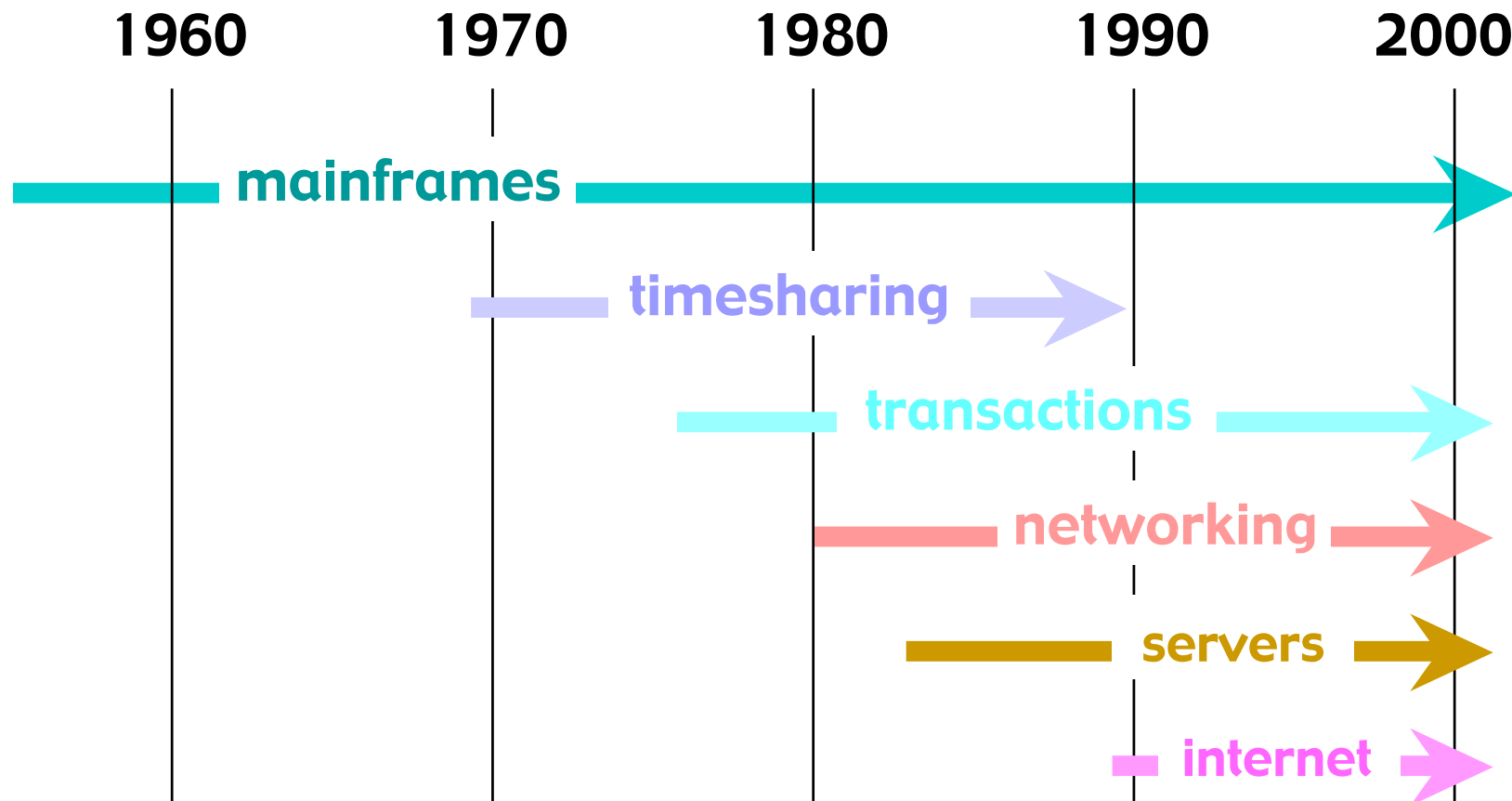
ondersteuning voor gedeelde *resources*:

-  **via gemeenschappelijk geheugen**
-  **via onderlinge communicatie**

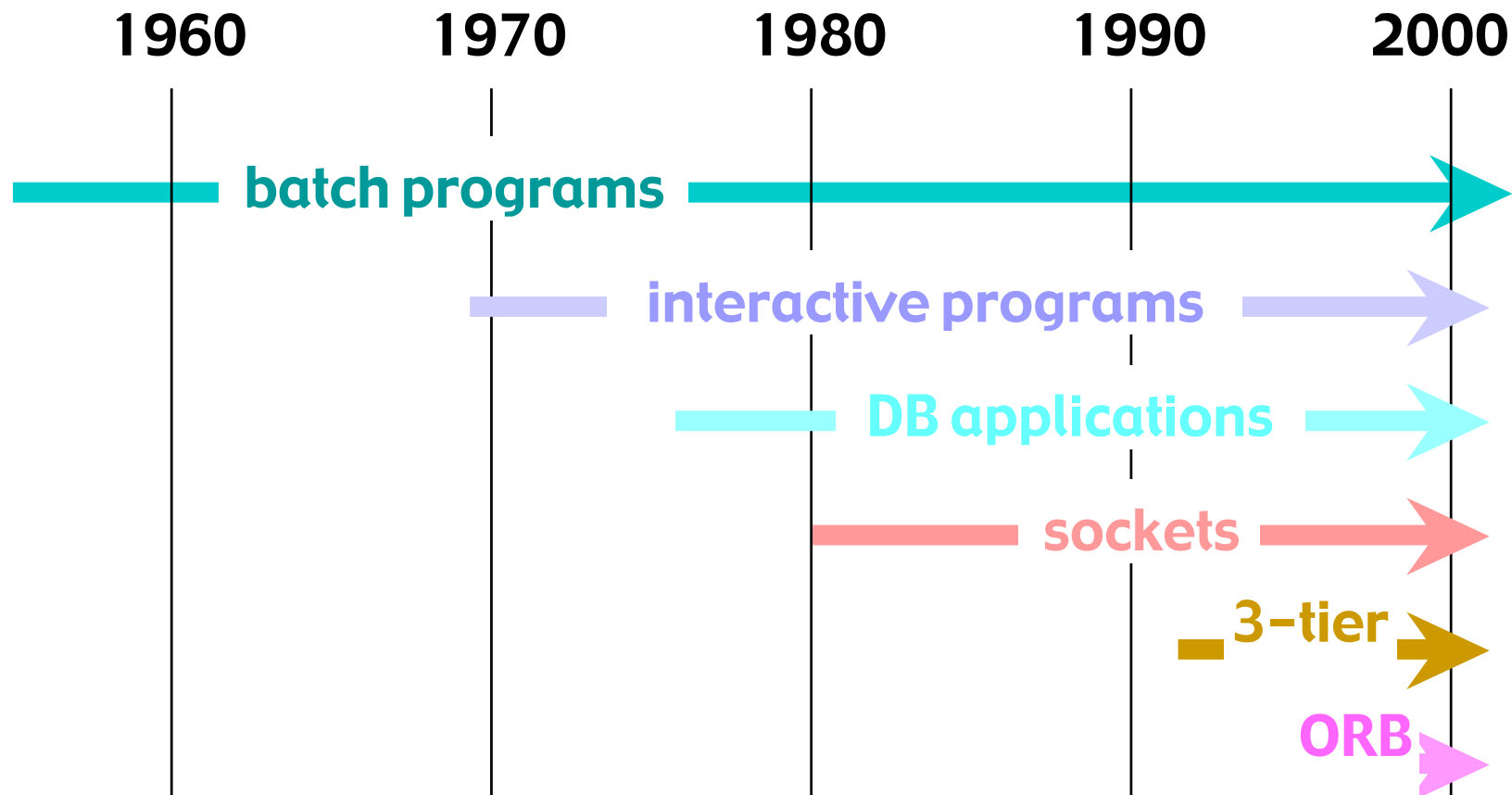
multi-processing

kritieke secties

Distributie: *timeline*



Distributie: *timeline*



Distributie: essentie

**Gedistribueerde verwerking
staat op het kruispunt van
(inter)netwerken en meer-
proces software**

**objecten zijn de
*enabling
technology***

Distributie: besluit

- ☑ ***incontournable* in moderne ICT**
- ☑ **na 40 jaar: geen oplossing**
- ☑ **conservatieve reacties (C#)**
- ☑ **nieuwe opportuniteiten**
- ☑ **historisch bewustzijn?**

Mobiliteit: gebruik

SMS

WAP

GSM

wireless
IP

PalmOS

Mobiliteit: *computing*

- hardware oplossingen**
- dikwijls autonoom**
- beperkte communicatie**
- simplistische interactie**

Mobiliteit: *embedded systems*

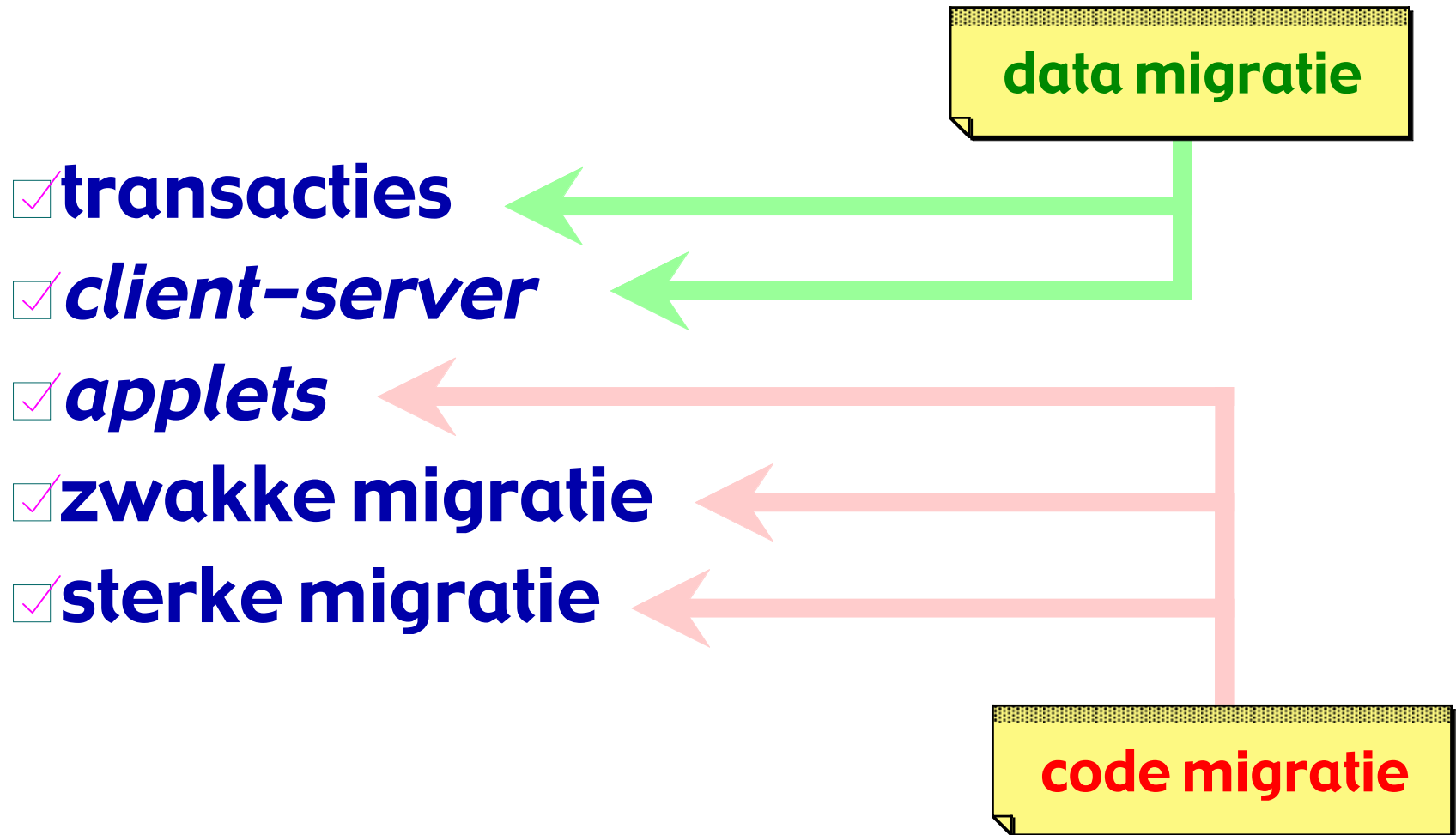
The logo for SEESCOA is a yellow rectangular box with a black border and a drop shadow. The text "SEESCOA" is written in blue, bold, uppercase letters inside the box.

SEESCOA

- software was beperkt door hardware**
- hardware schaal is enorm gegroeid**
- software engineering?**
- objectgerichtheid?**
- Java!**

<http://www.cs.kuleuven.ac.be/cwis/research/distrinet/projects/SEESCOA/>

Mobiliteit: migratie



Mobiliteit: data migratie

- ☑ **vaste code duplicatie**
- ☑ **centraal beheer van data**
- ☑ **tijdelijke data duplicatie**
- ☑ **gaat gepaard met data *locking***
- ☑ **dus: transacties**

**TP-monitors
bestaan al
30+ jaar ...**

Mobiliteit: code migratie

- ☑ **tijdelijke code duplicatie**
- ☑ **variabel beheer van data**
- ☑ **variabele data duplicatie**
- ☑ **gaat gepaard met object *locking***
- ☑ **dus: *concurrency***

Mobiliteit: *applets*

- centraal code beheer**
- Java beveiliging**
- tijdelijke data duplicatie (R/W)**
- tijdelijke code duplicatie (R/O)**
- application services?***

Mobiliteit: zwakke migratie

- ☑ **migratie van objecten**
- ☑ **uniforme, verdeelde vertolking**
- ☑ **migratie = stop/start**
- ☑ **Java: *marshalling***
- ☑ **Java: *reflection API***

cfr. *EMOOSE*

Mobiliteit: sterke migratie

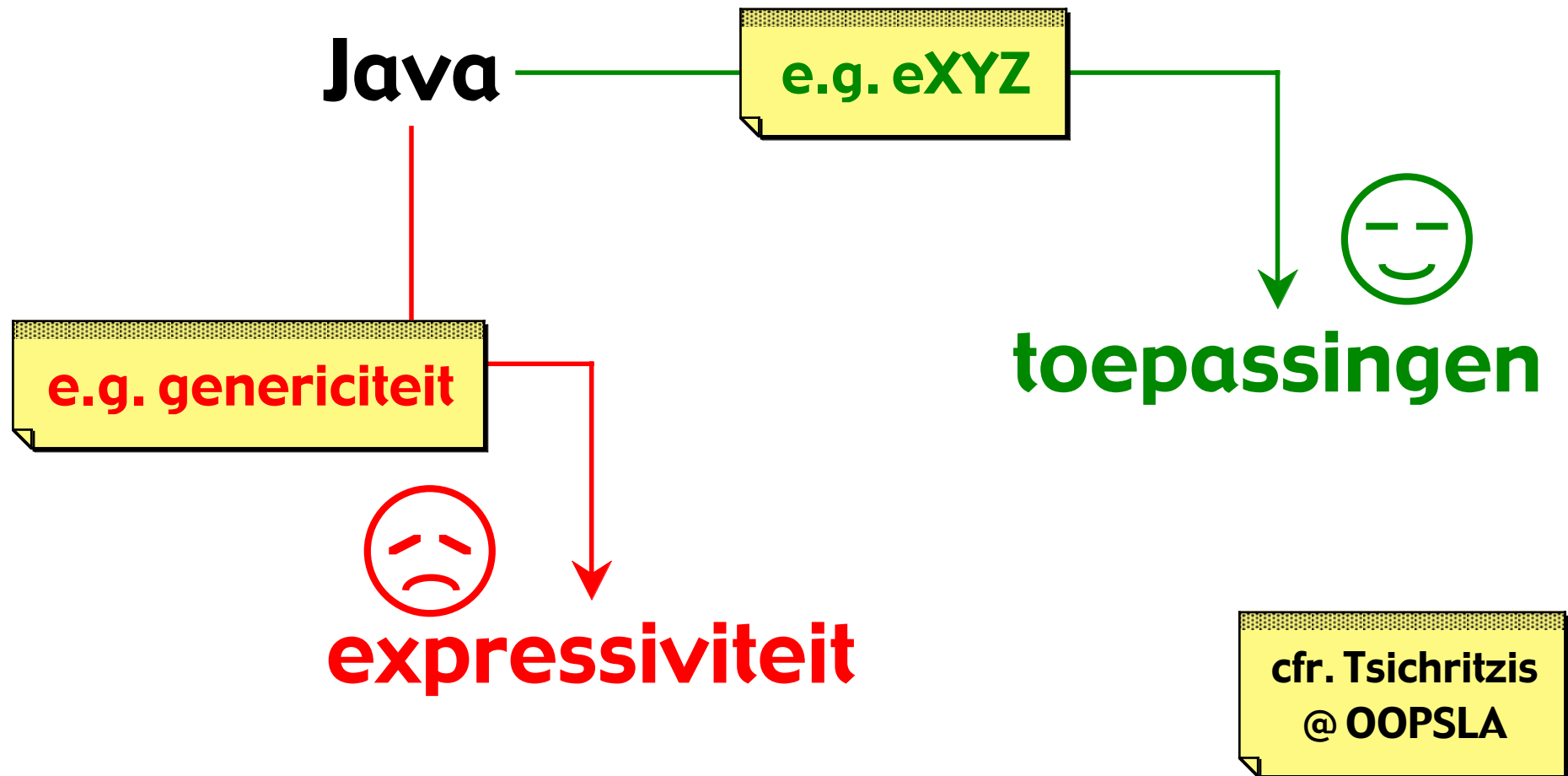
- ✓ migratie van objecten
- ✓ uniforme, verdeelde vertolking
- ✓ migratie = *hold/continue*
- ✓ code introspectie
- ✓ reflectieve virtuele machine

cfr. *EMOOSE*

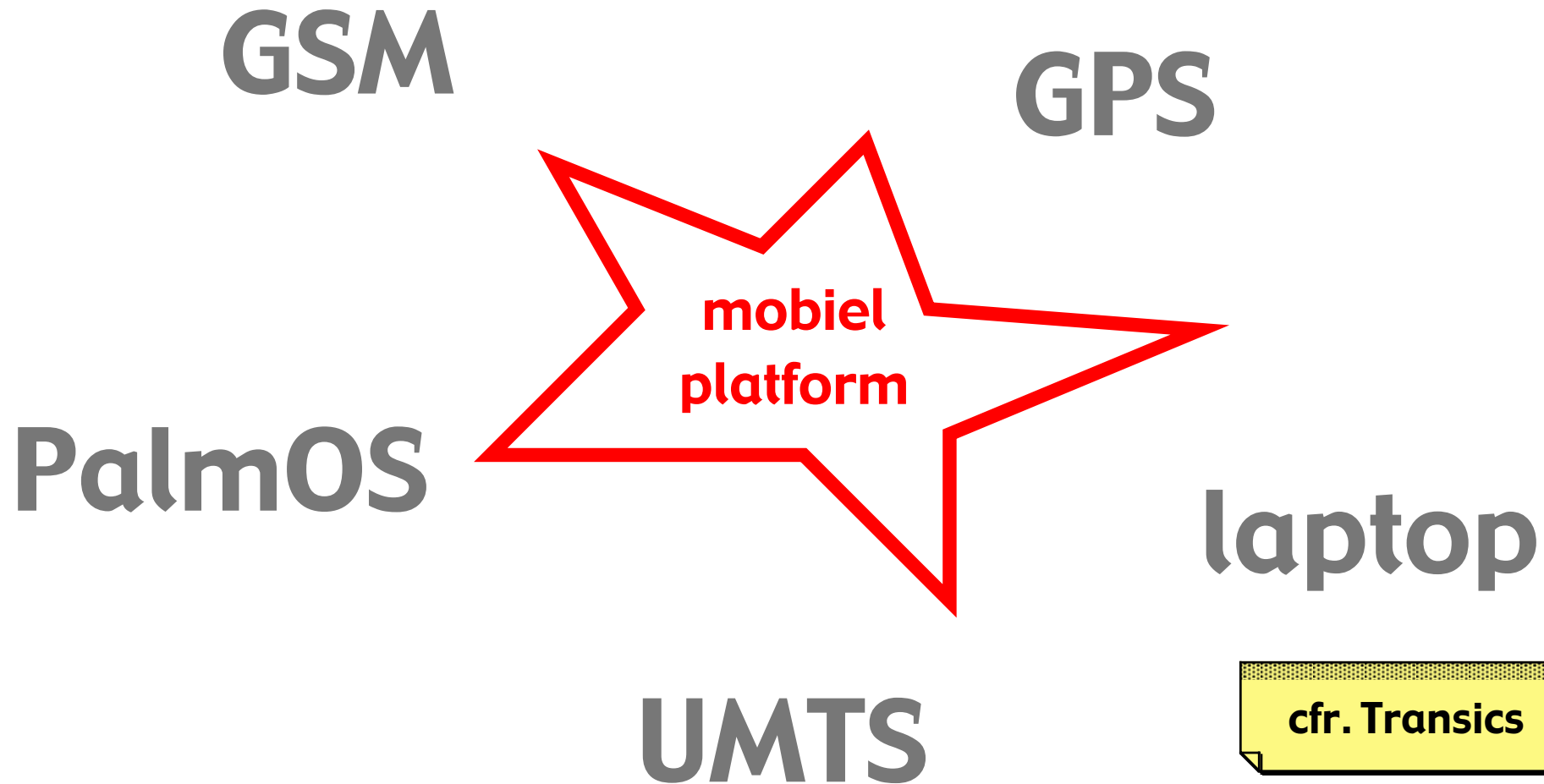
Mobiliteit: *besluit*

- ☑ **mobiele hardware**
- ☑ **statische toepassingen**
- ☑ **communicatie *bottleneck***
- ☑ **weinig focus op innovatie**
- ☑ **opportunistisch**

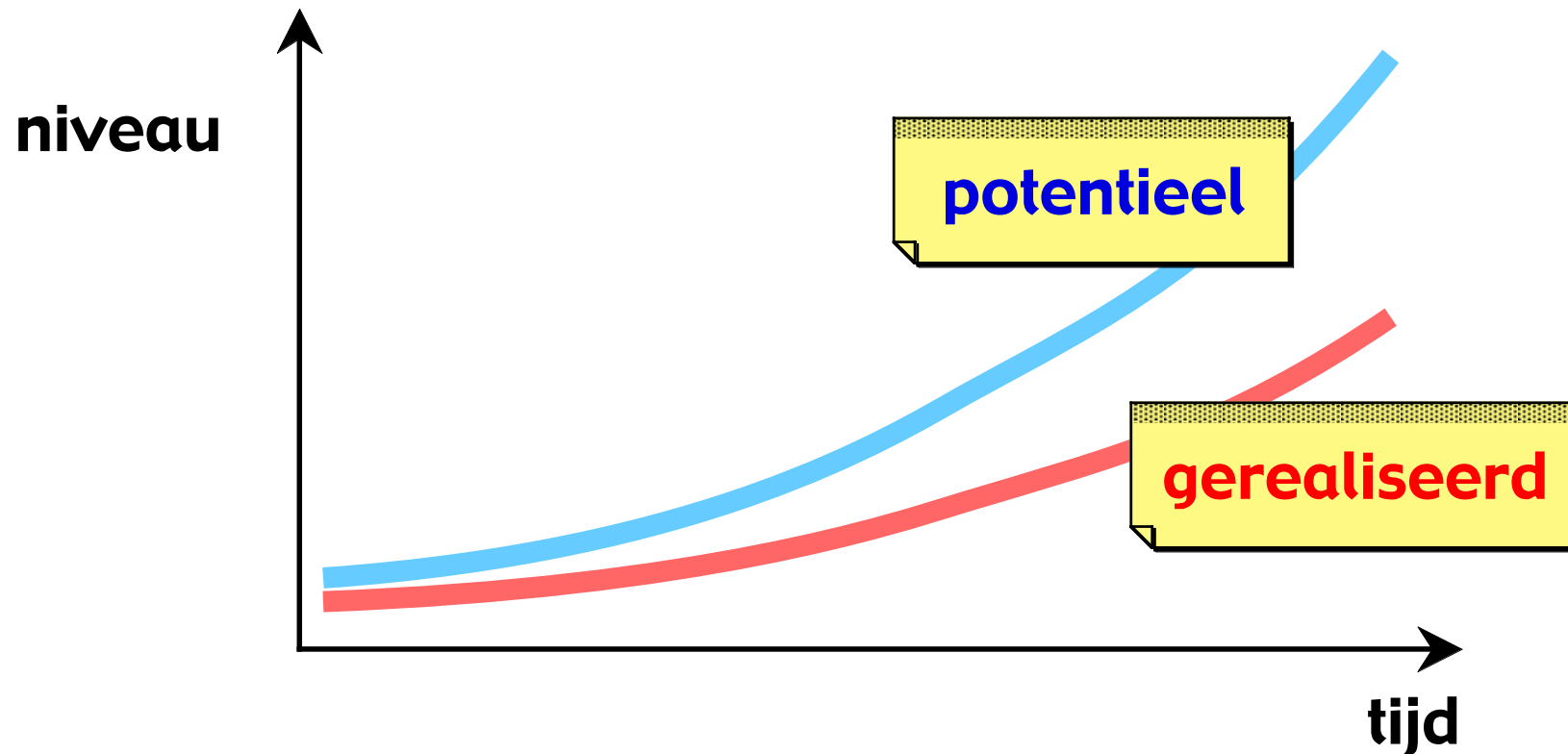
Evolutie: Java



Evolutie: technologie

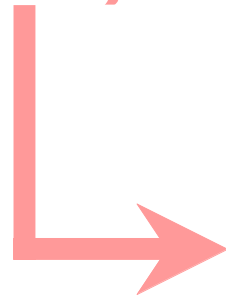


Evolutie: applicaties

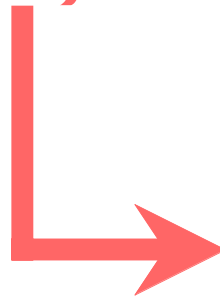


Evolutie: *bottleneck*

gedistribueerde
mobiele objecten



expressieve
ontwikkelsystemen



expertise +
methodiek

Besluit

Voorzienbare toekomst:

- hardware >> software
- "*thin*" toepassingen

Voorzienbare noden:

- methode/*tools*
- opleiding
- bewustzijn