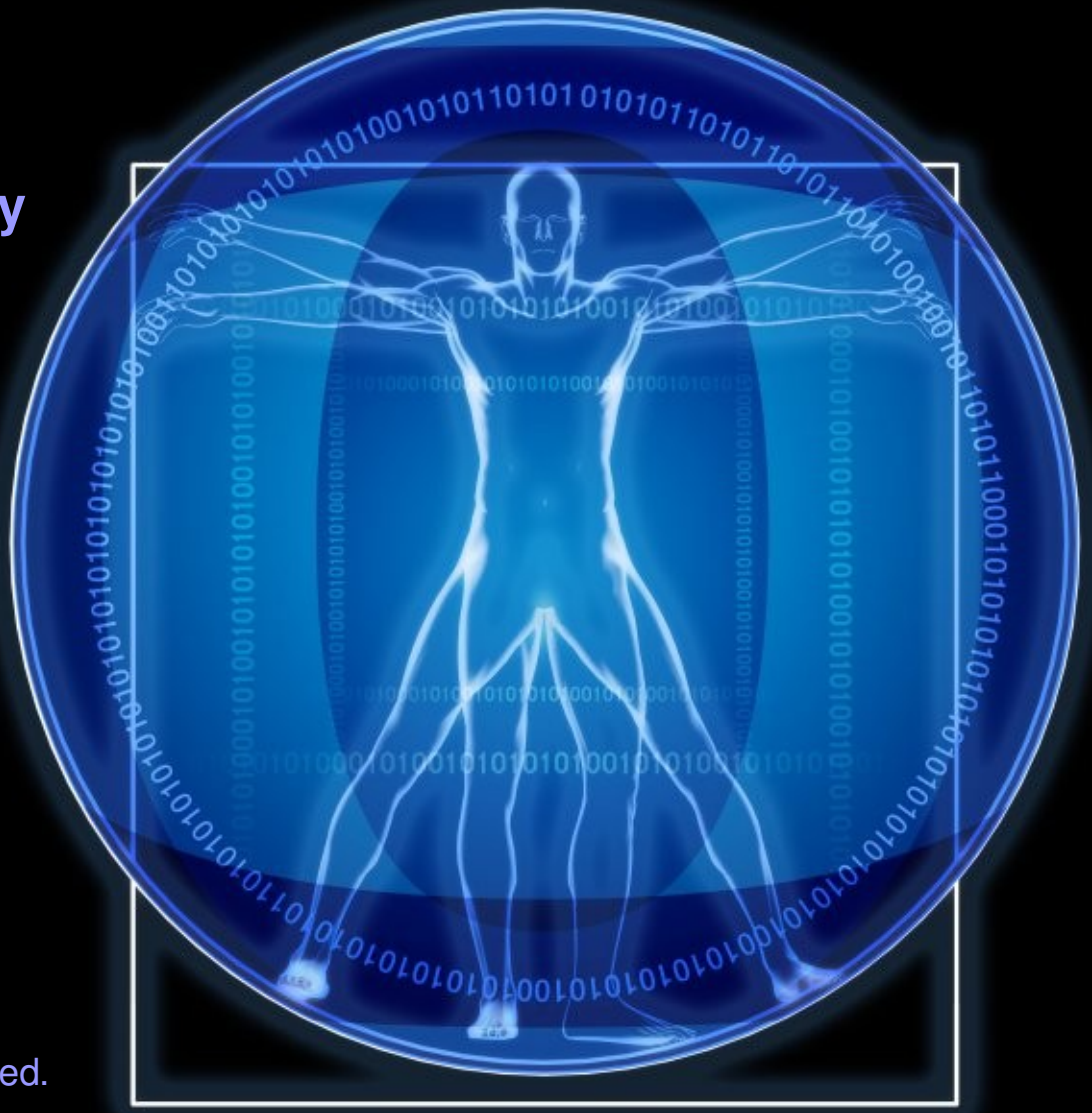
Cybernetics Oriented Programming (CYBOP)

An Investigation on the Applicability of Inter-Disciplinary Concepts to Software System Development



LinuxTag 2007, Berlin, Germany

Dr. Christian Heller

http://www.cybop.net <christian.heller@tuxtax.de>

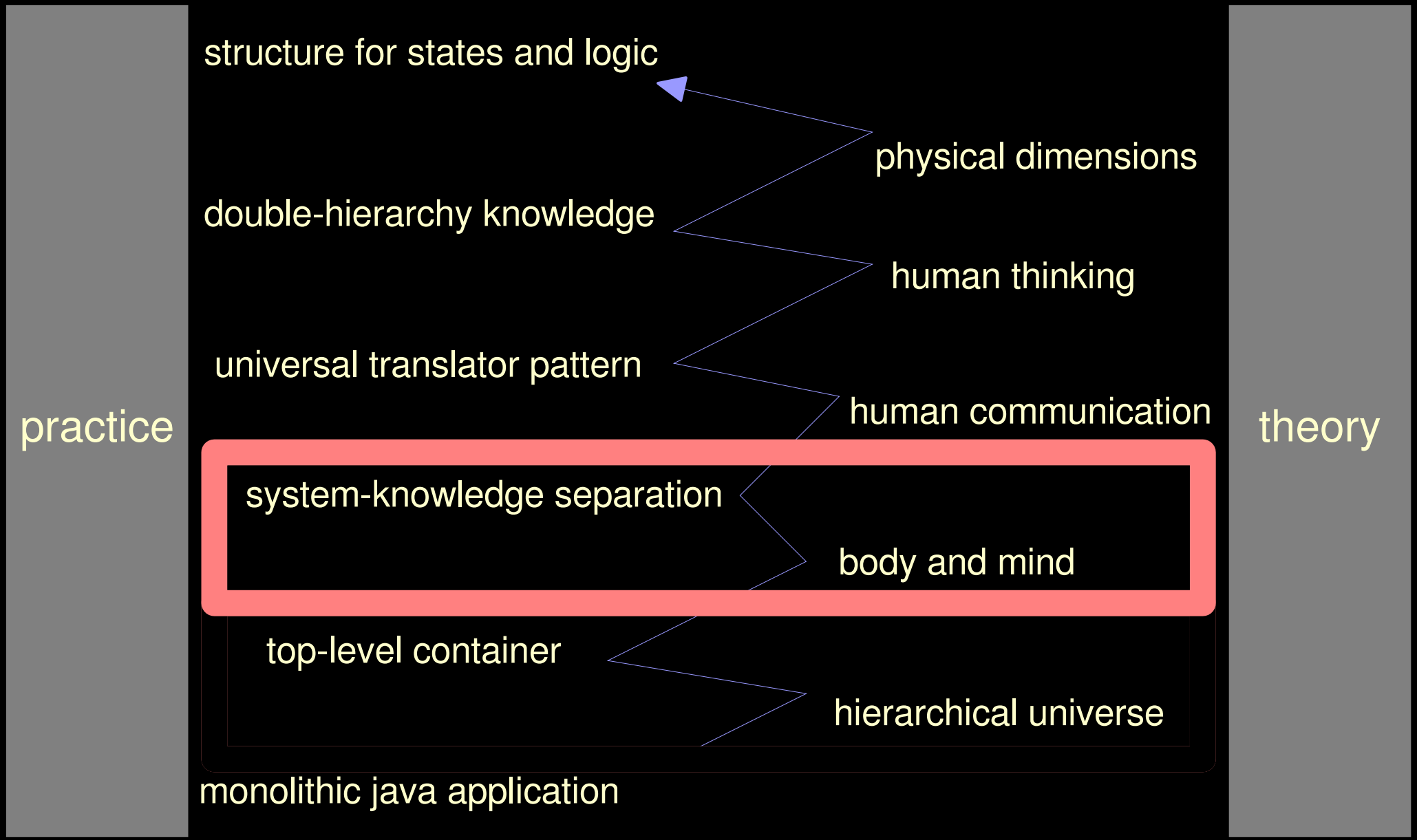
knowledge

human being

software

hardware

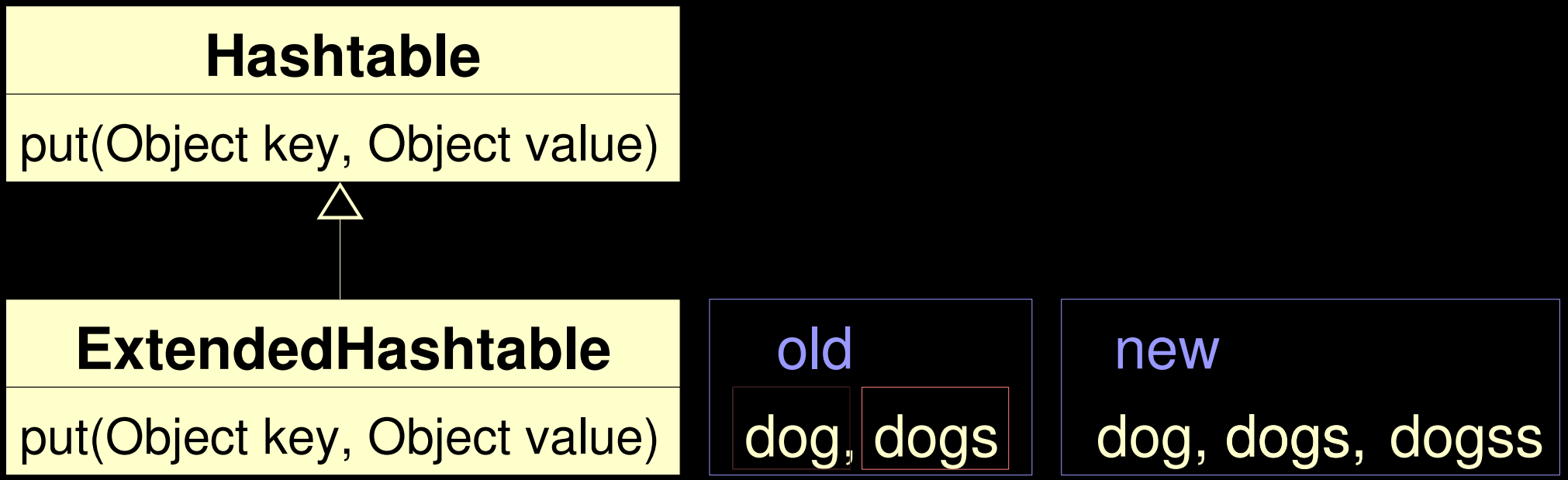
information	fact or message with recognisable news in semantic context
data	(machine-readable) characters / numbers that may contain information
knowledge	structured data which are inter-related (associated)



constructive development – complexity – example

- bundling of attributes and methods – coupling, no flexibility
- reflective meta architectures – bidirectional dependencies
- bidirectional dependencies – complexity, circular references
- global / static data access – untraceable data manipulation
- container inheritance – falsified container content

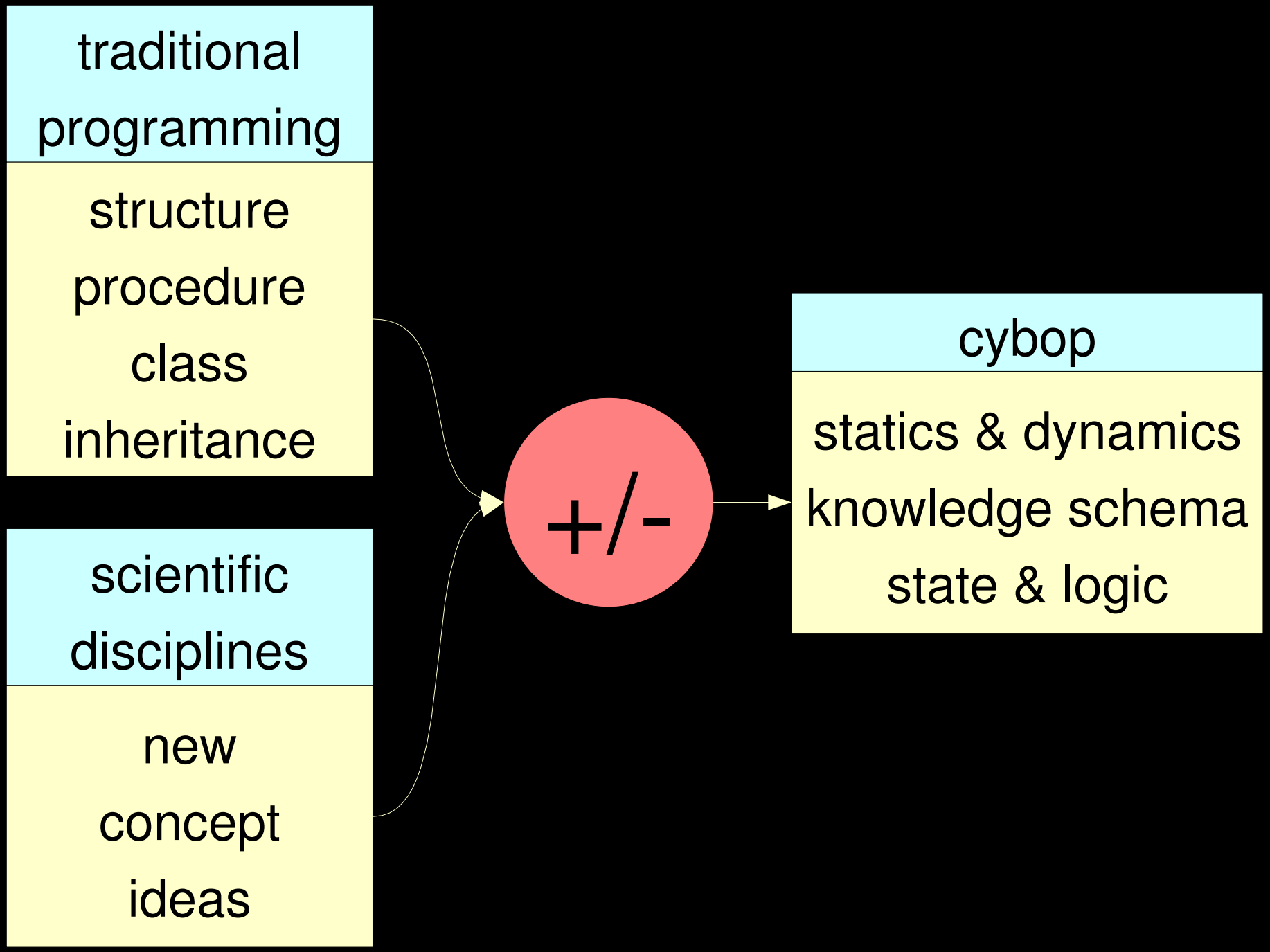
problems – more in cybop book




```
super.put(key, value);
super.put(key + “s”, value);
```

copy

falsifying container inheritance [iaq, dr. norvig]


traditional programming
structure
procedure
class
inheritance
scientific disciplines
new
concept
ideas
+/-
cybop
statics & dynamics
knowledge schema
state & logic

cybernetics (kybernetes = steersman)

- science of information and control
- in living things or machines (norbert wiener)

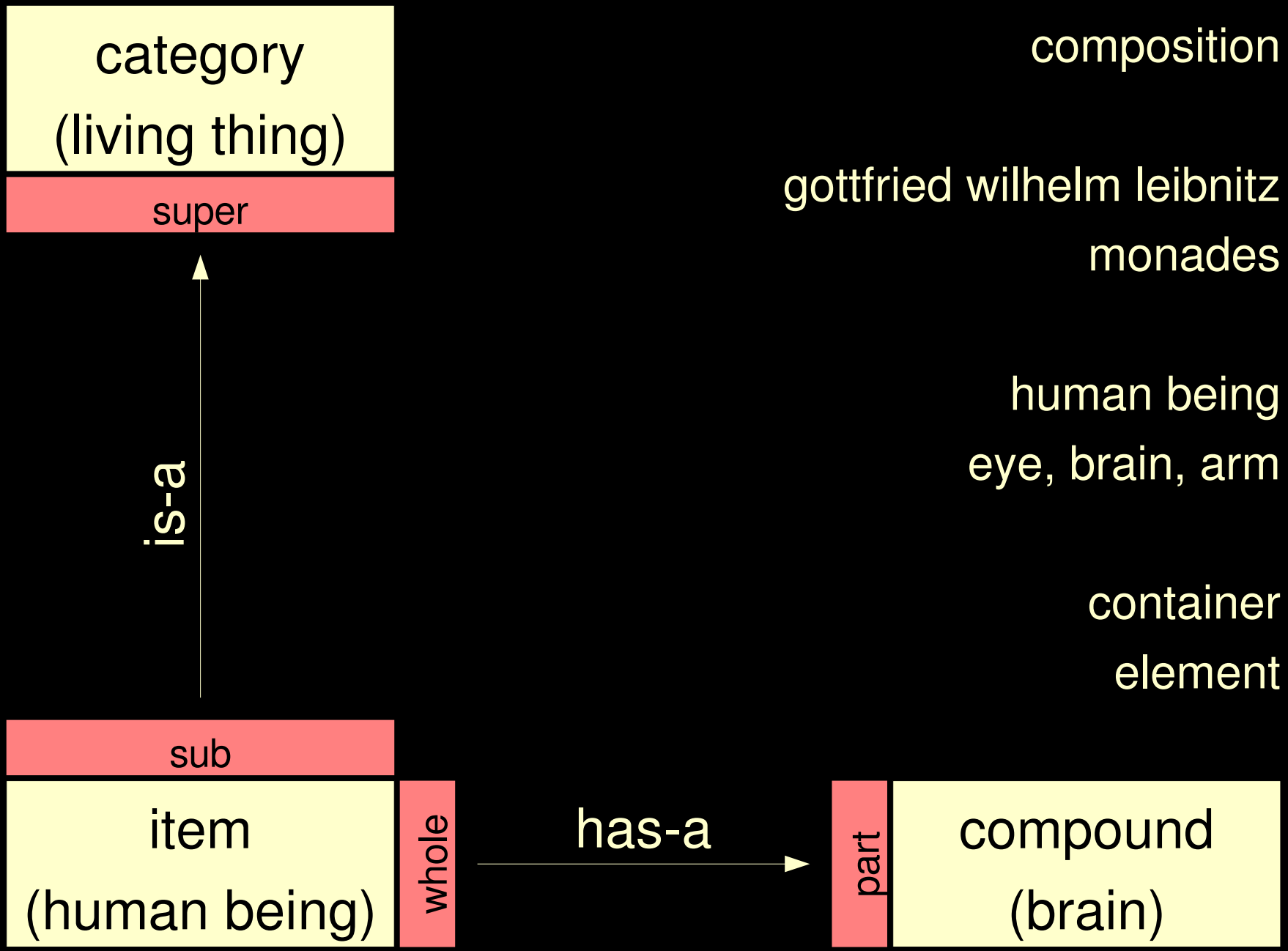
bionics (bio-cybernetics)

- biological principles applied to
- study and design of engineering systems

relation

- software engineering = systems engineering
- system as a whole gains in importance
- biological / human → software system
- physical brain: neural network
- logical mind: concepts

inter-disciplinary
CYBOP

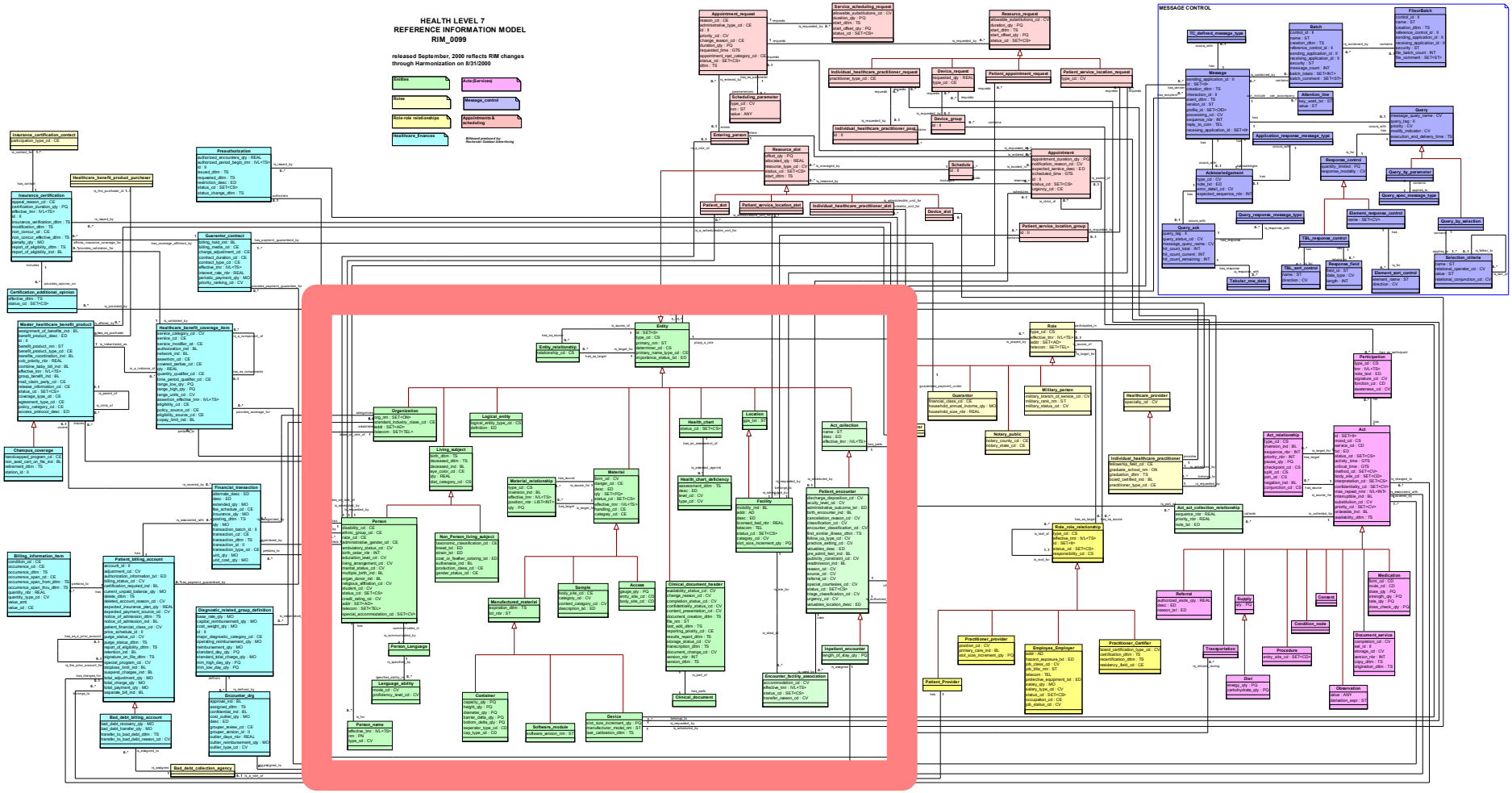


composition

gottfried wilhelm leibnitz
monades

human being
eye, brain, arm

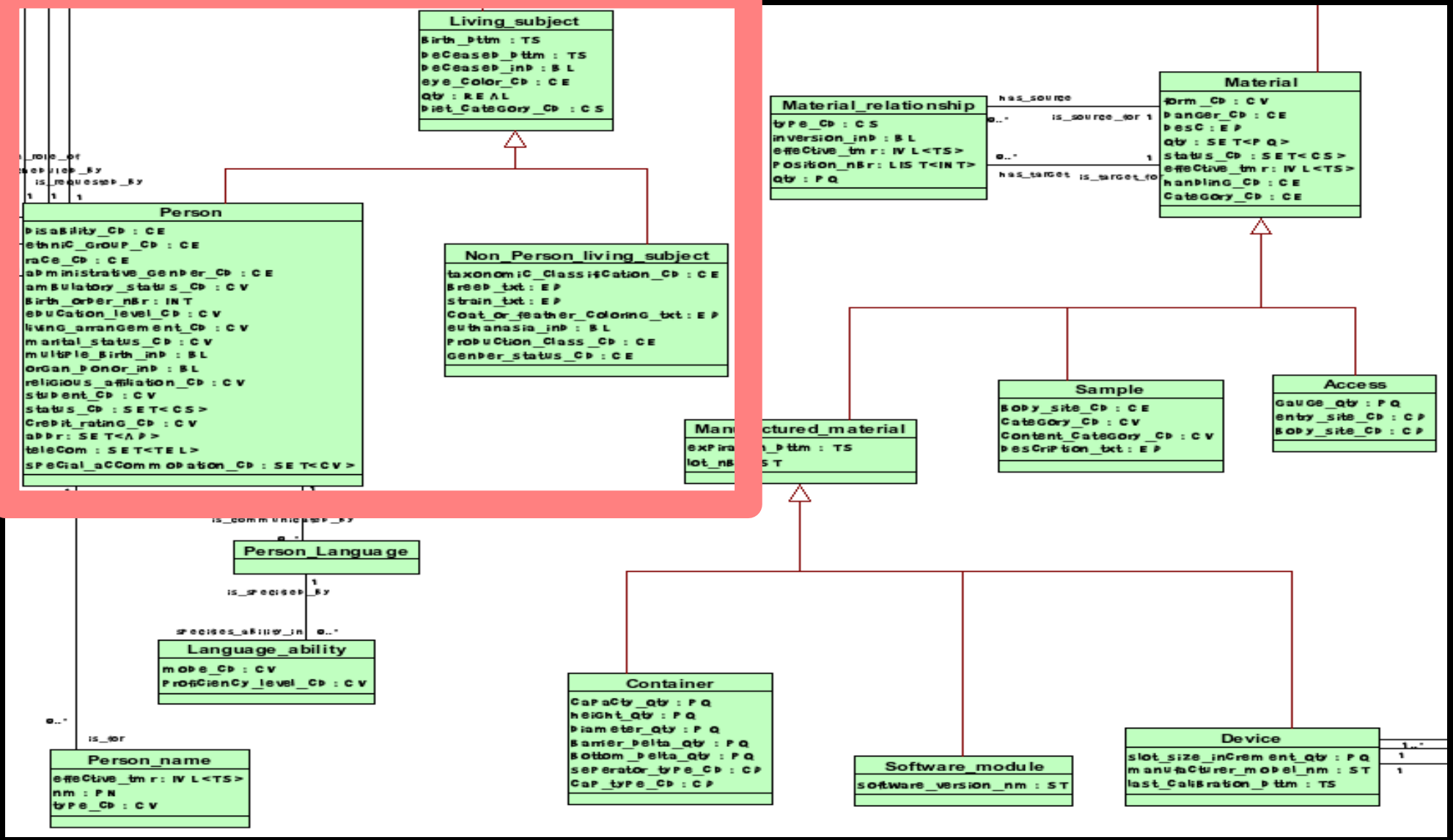
container
element



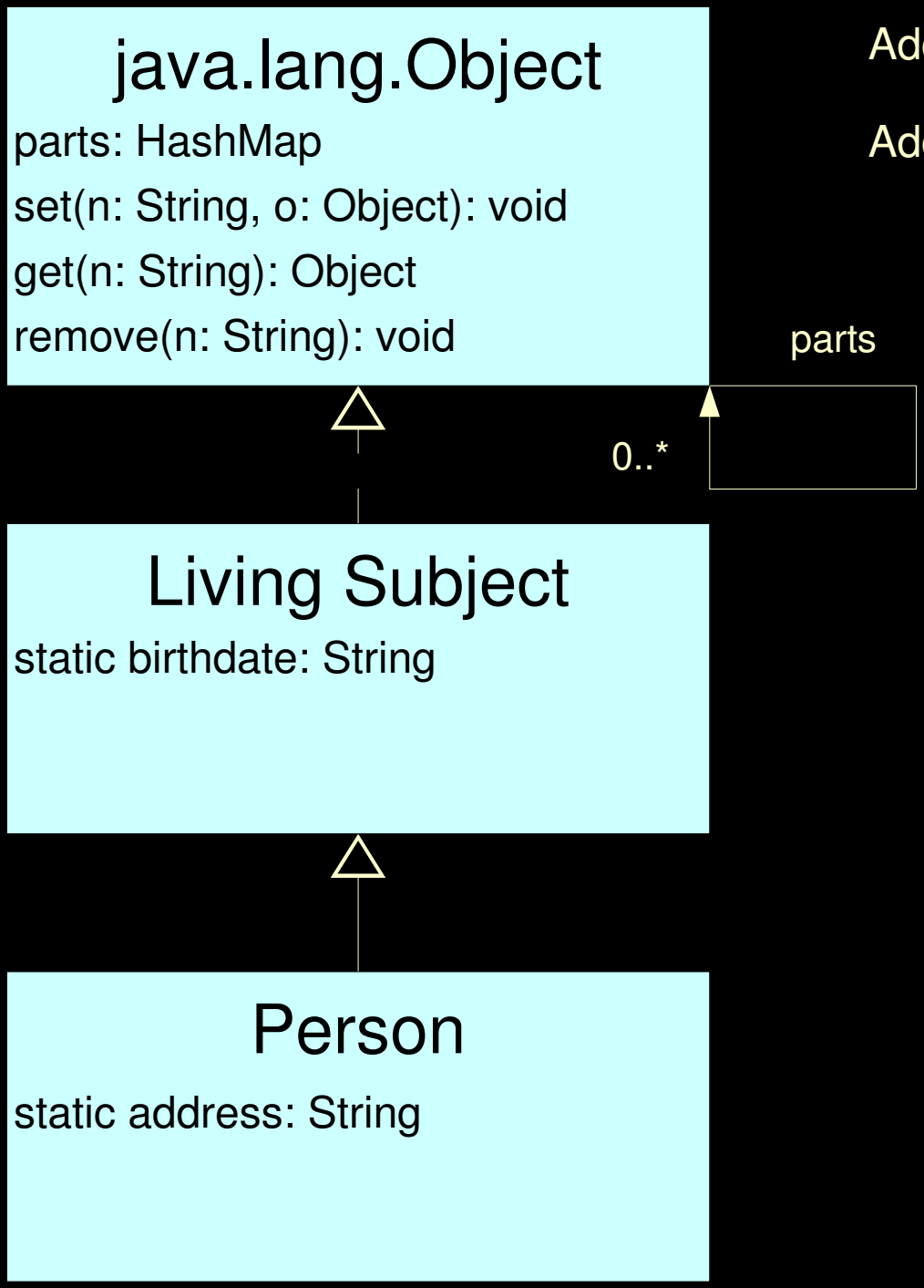
Health Level 7 - RIM
Reference Information Model

scheduling		message
finance	role	service
entity	role-role-relation	

statics and dynamics



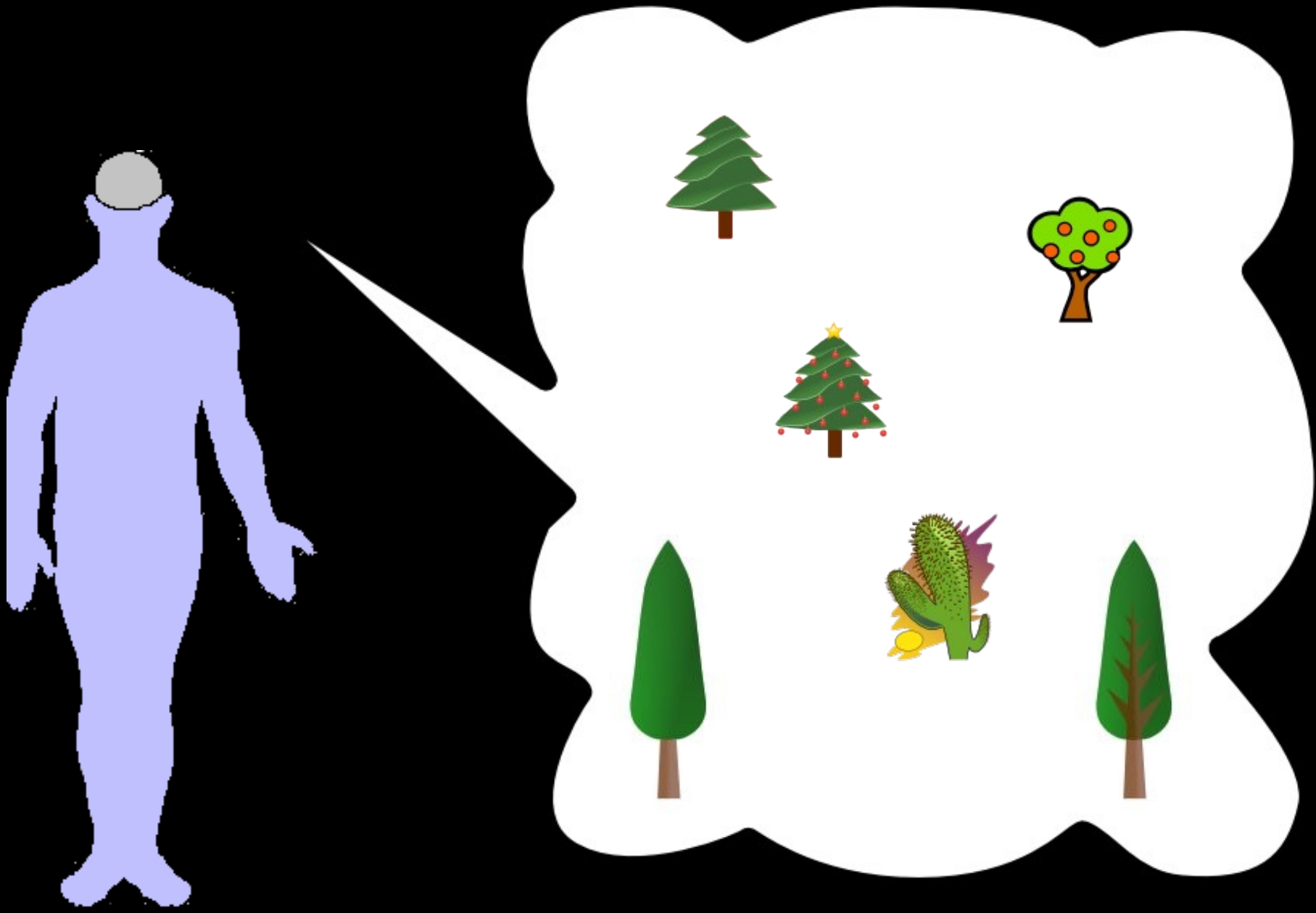
- discrimination
- categorisation
- composition




```
Address a = (Address) get("address");

Address a = (Address) get(Person.address);
```

extract knowledge



system owns knowledge (also: biological cell + dna)

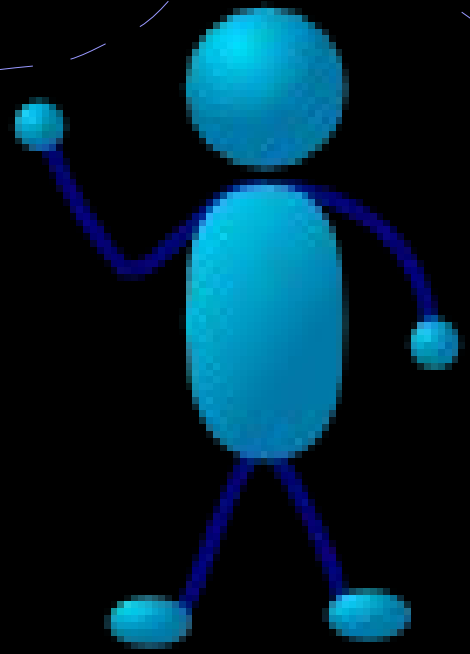
properties

happy, sad, aggressive

black, white

shape, size

smell



head, eyes, ears, hair

state structure

arms, legs

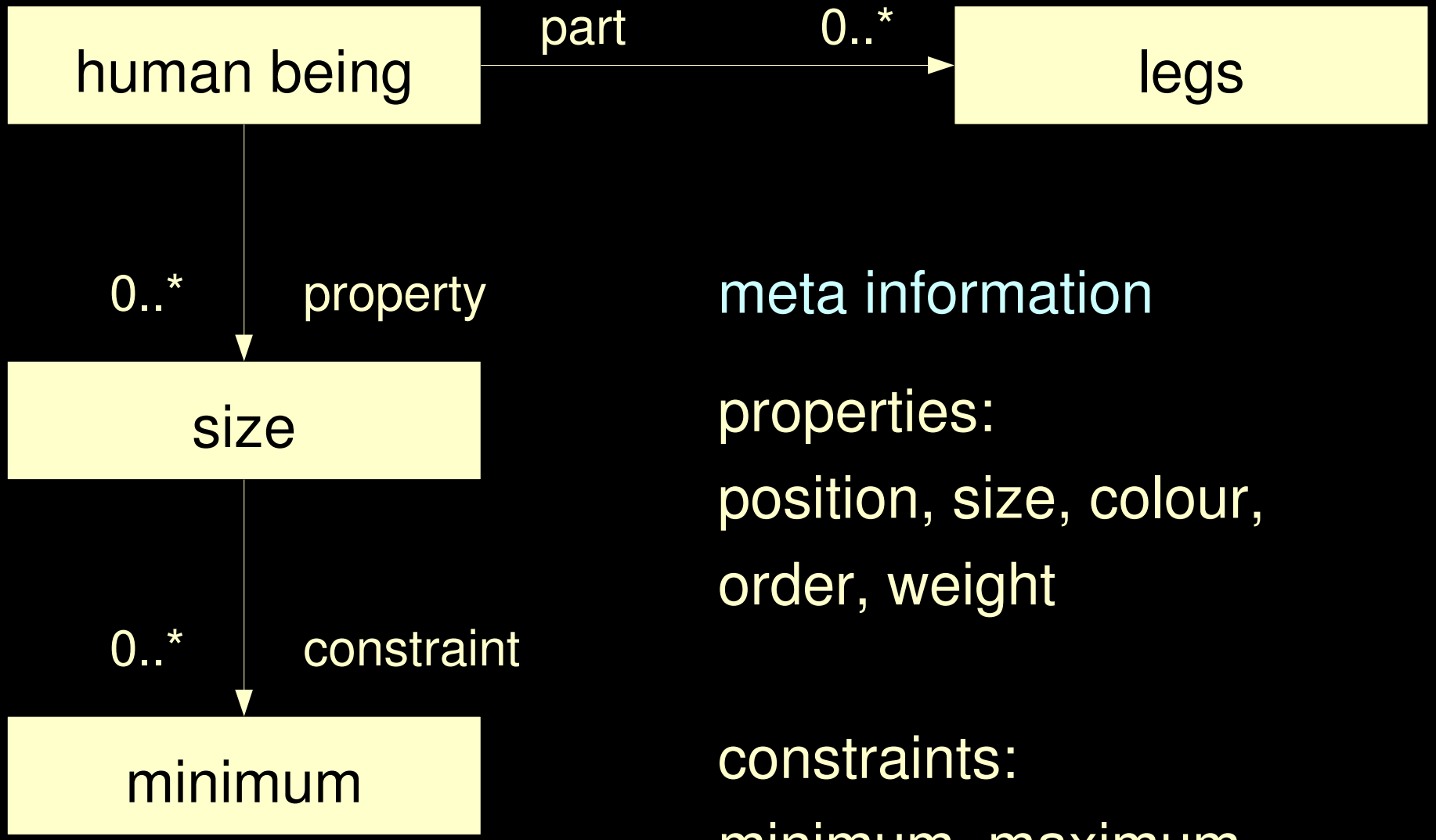
food, book

clothes, shoes, hat

external concepts

walk, run, limp

change logic



meta information

properties:

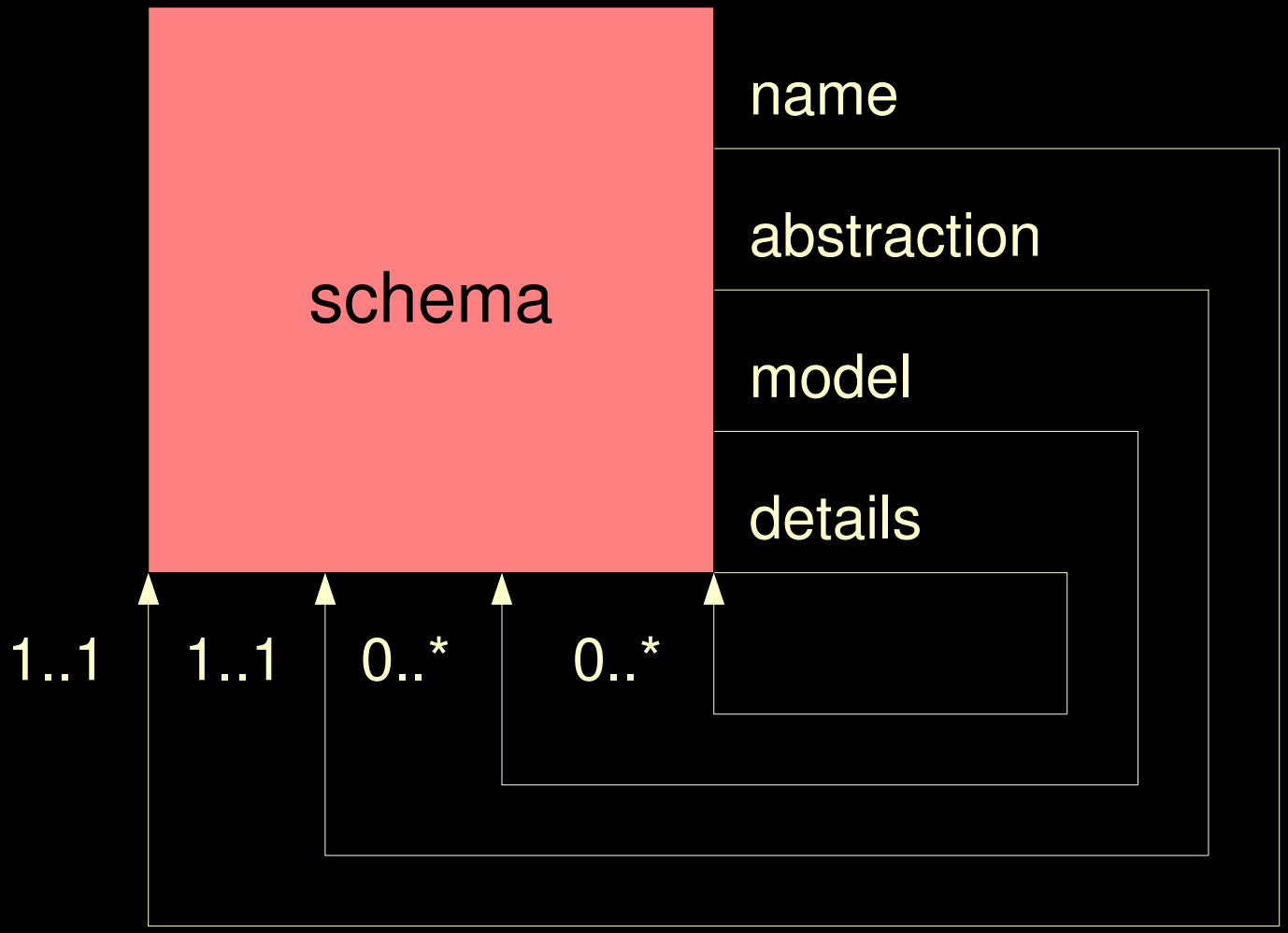
position, size, colour,

order, weight

constraints:

minimum, maximum,

double hierarchy – in space, time etc.

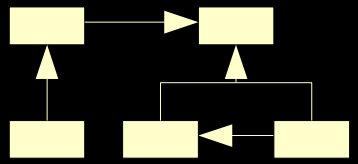


item
+ category
+ compound
= schema

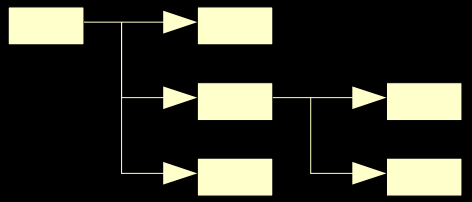
relations
unidirectional

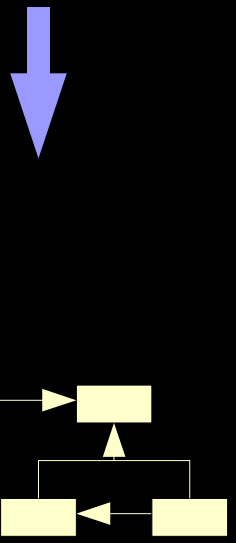
traditional

cybop

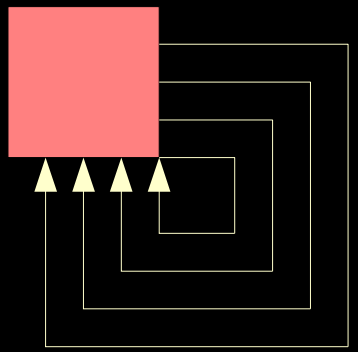


program
structure





runtime
structure

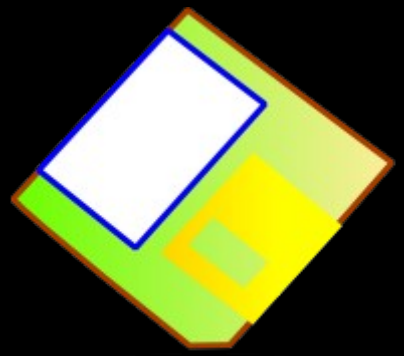


universal memory structure – flexibility

interact

presentation client

read / write



configure

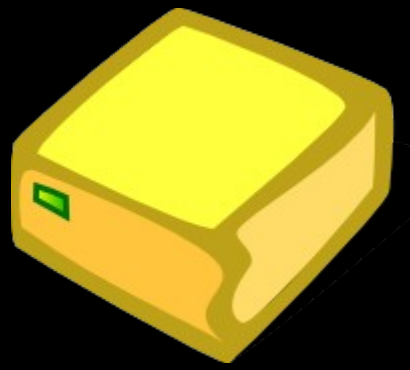
application server

read / write

administer

database server

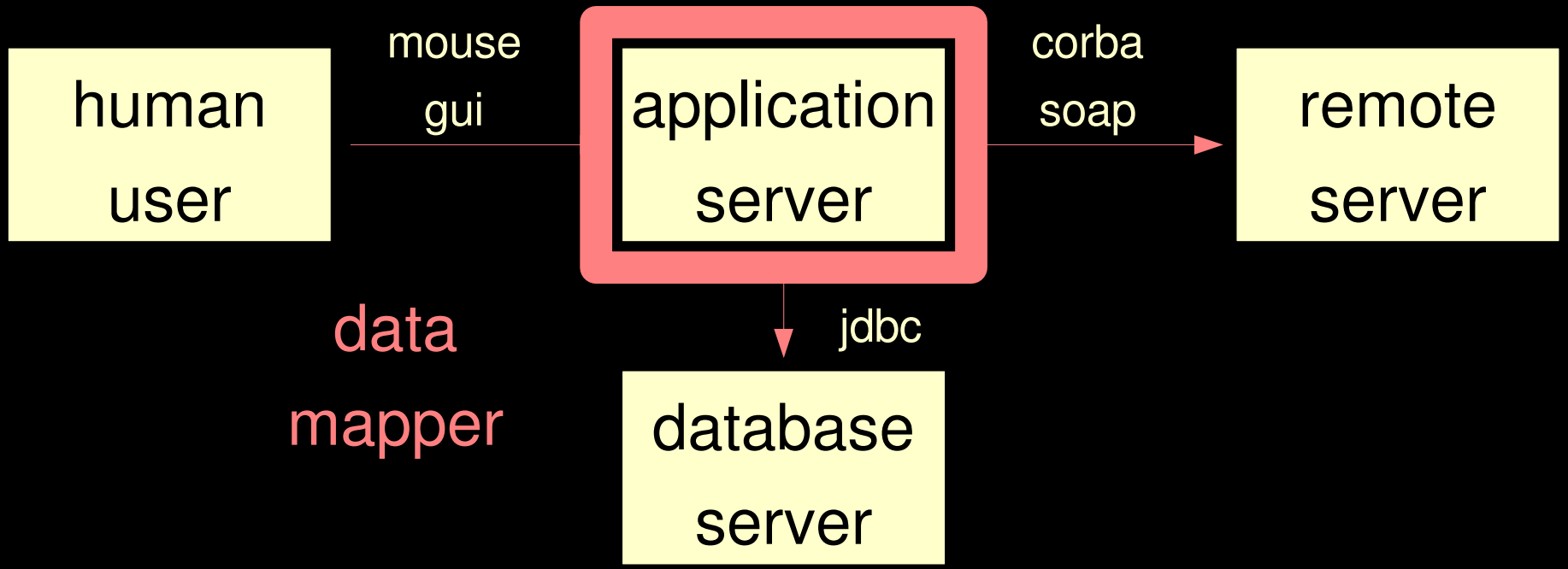
read / write



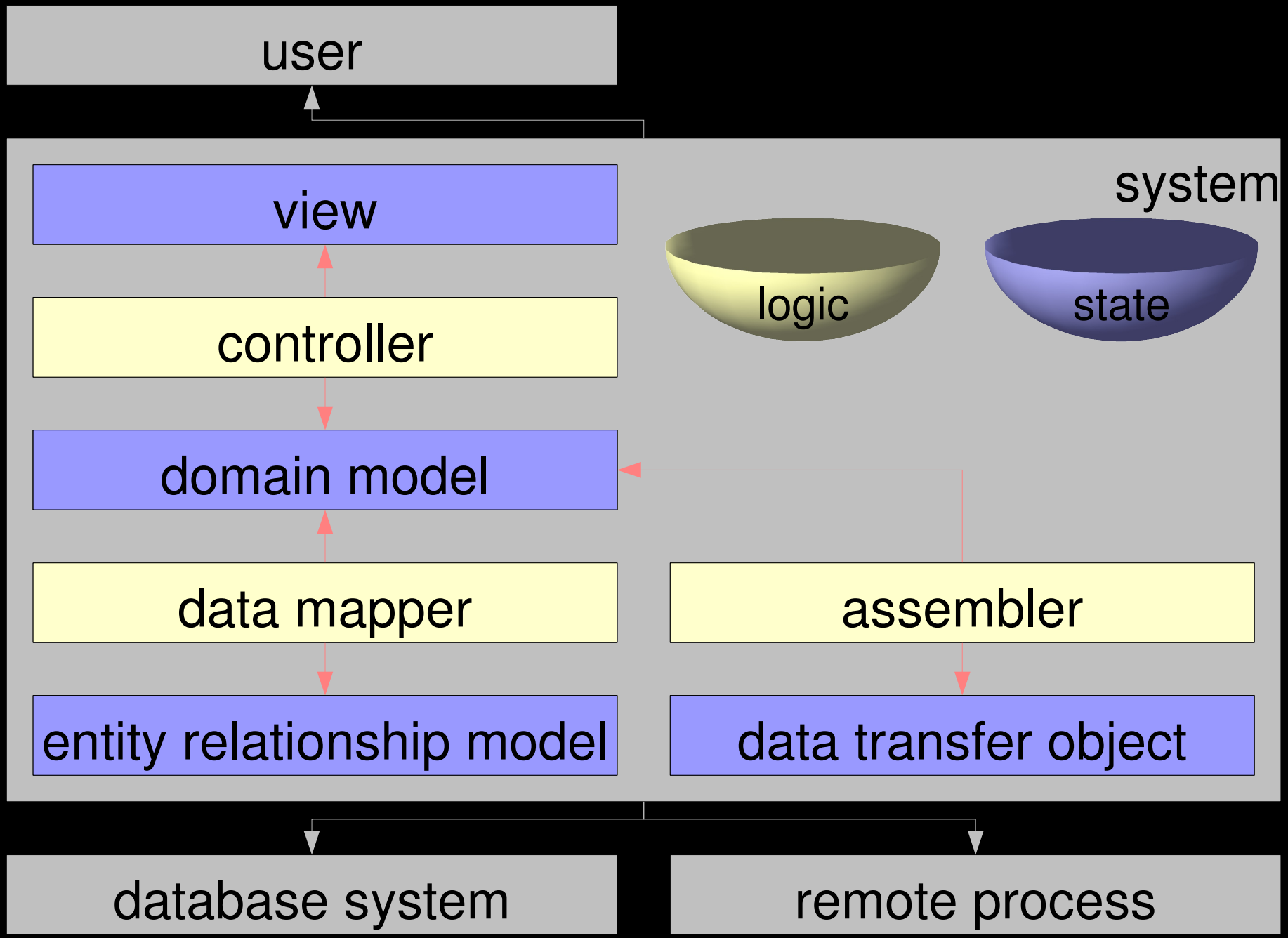
misleading tiers – inflexible software architecture

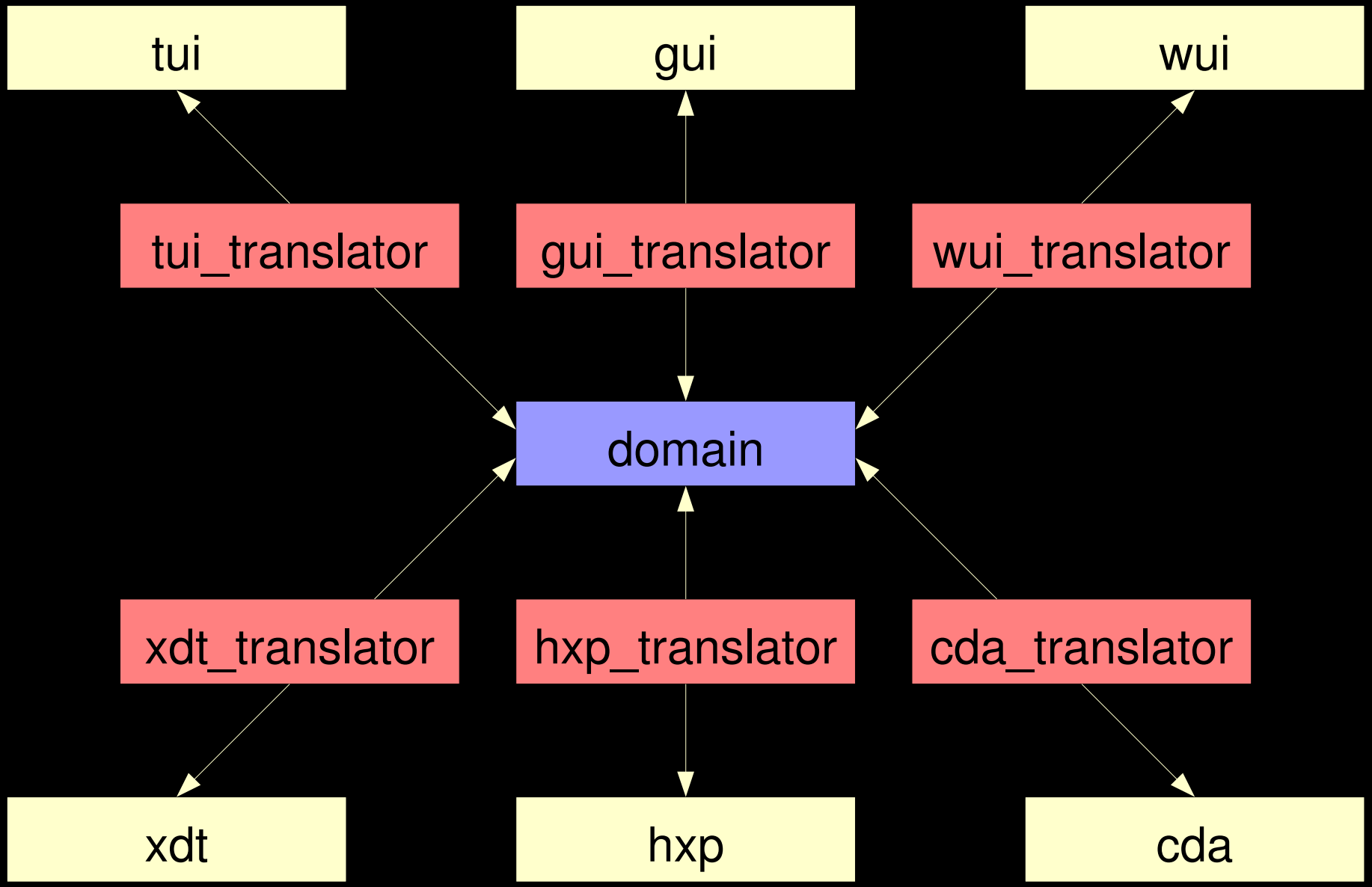
model view controller

data transfer object



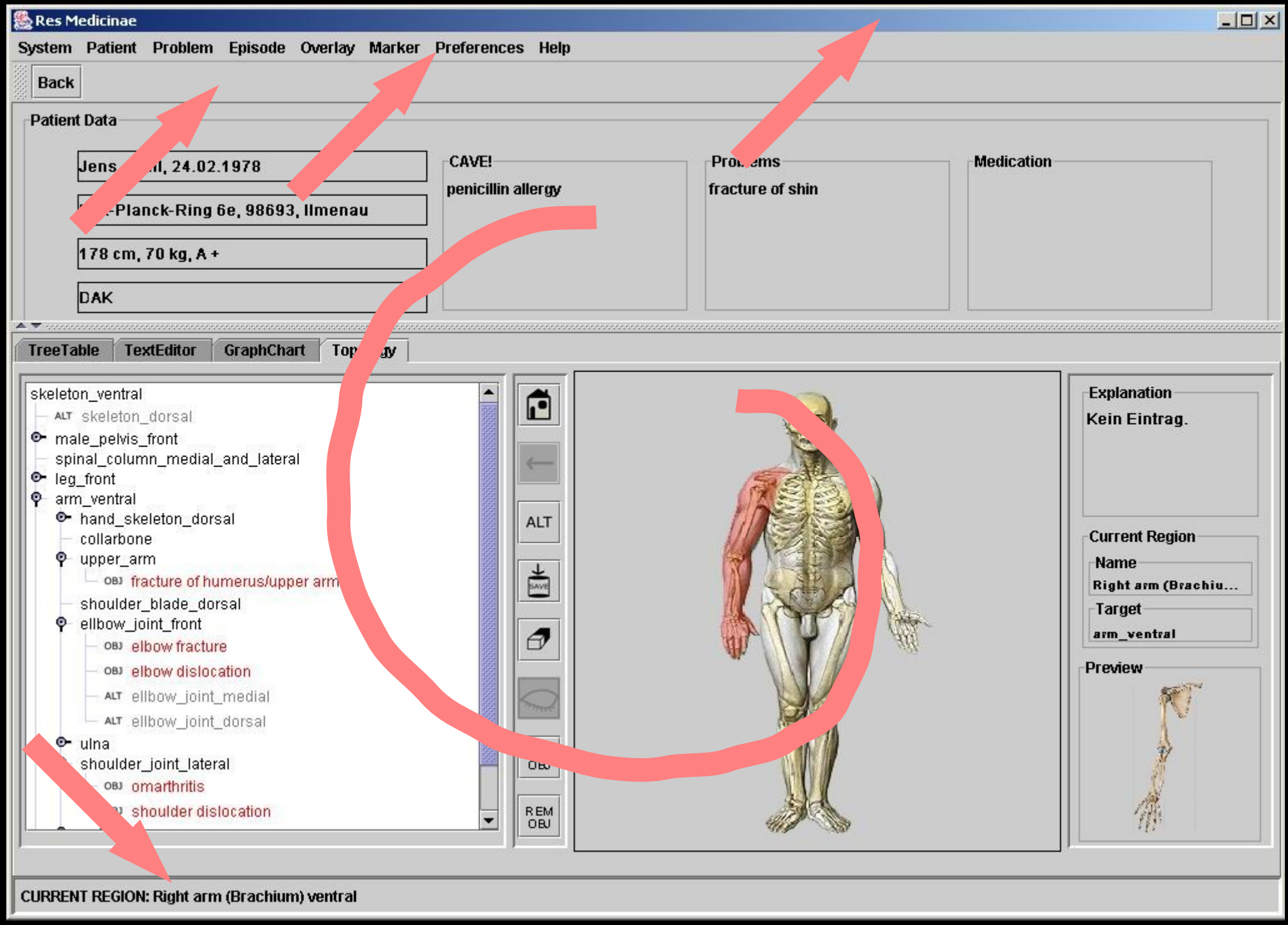
communication patterns


user
system
view
logic
state
controller
domain model
data mapper
assembler
entity relationship model
data transfer object
database system
remote process



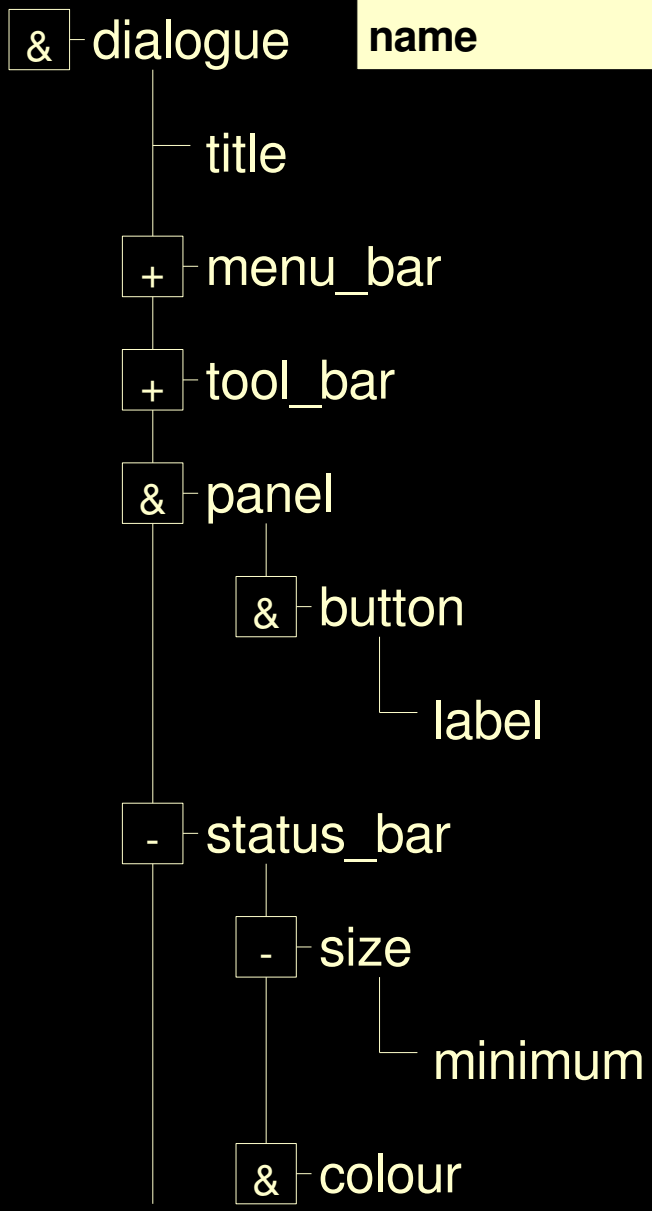
star-like (not layer-like) translator architecture

cybol language



```
<model>
  <part name="title" channel=”inline” abstraction=”character” model="Res Medicinae">
    <property name=”size” channel=”inline” abstraction=”integer” model=”600,20,1”>
      <constraint name=”minimum” channel=”inline” abstraction=”integer” ... />
      <constraint name=”maximum” channel=”inline” abstraction=”integer” ... />
    </property>
    <property name=”position” channel=”inline” abstraction=”integer” model=”0,0,0”/>
    <propertyname="layout"channel="inline"abstraction="character"model="coordinate"/>
    <property name="colour" channel="inline" abstraction="character" model="blue"/>
  </part>
  <part name="menu_bar" ...
  <part name="tool_bar” ...
  <part name="content” ...
  <part name="status_bar” channel=”file” abstraction="compound" model="status.cybol”>
    <property name="layout" channel="inline"abstraction="character"model="compass"/>
    <property name="cell" channel="inline" abstraction="character" model="north"/>
```

cybol tags + attributes, abstraction principles

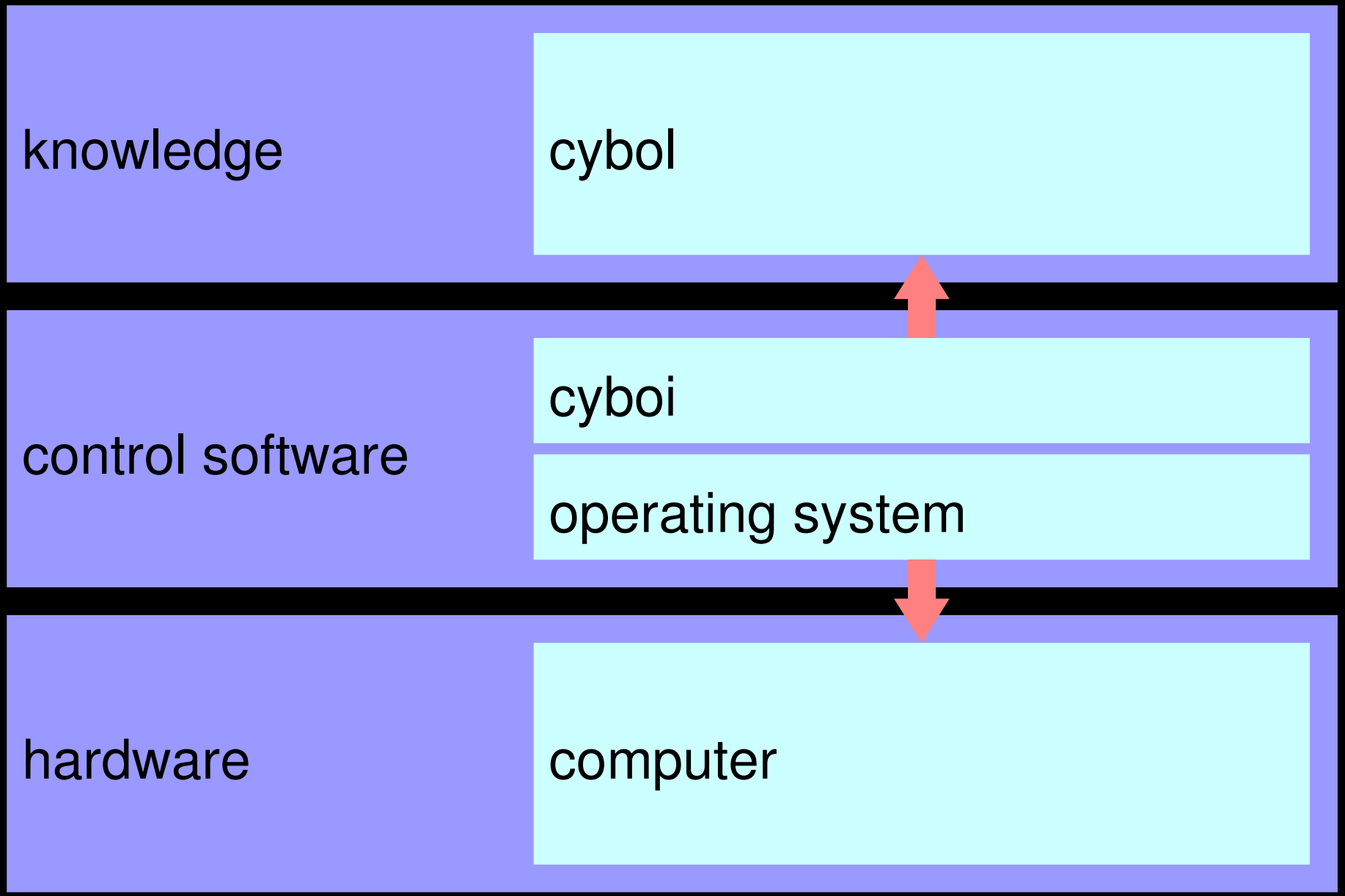


name	channel	abstraction	model
	inline	character	Res Medicine
	file	compound	/resmedicinae/gui/menu_bar.cybol
	file	compound	/resmedicinae/gui/tool_bar.cybol
	file	compound	/resmedicinae/gui/panel.cybol
	file	compound	/resmedicinae/gui/status_bar.cybol

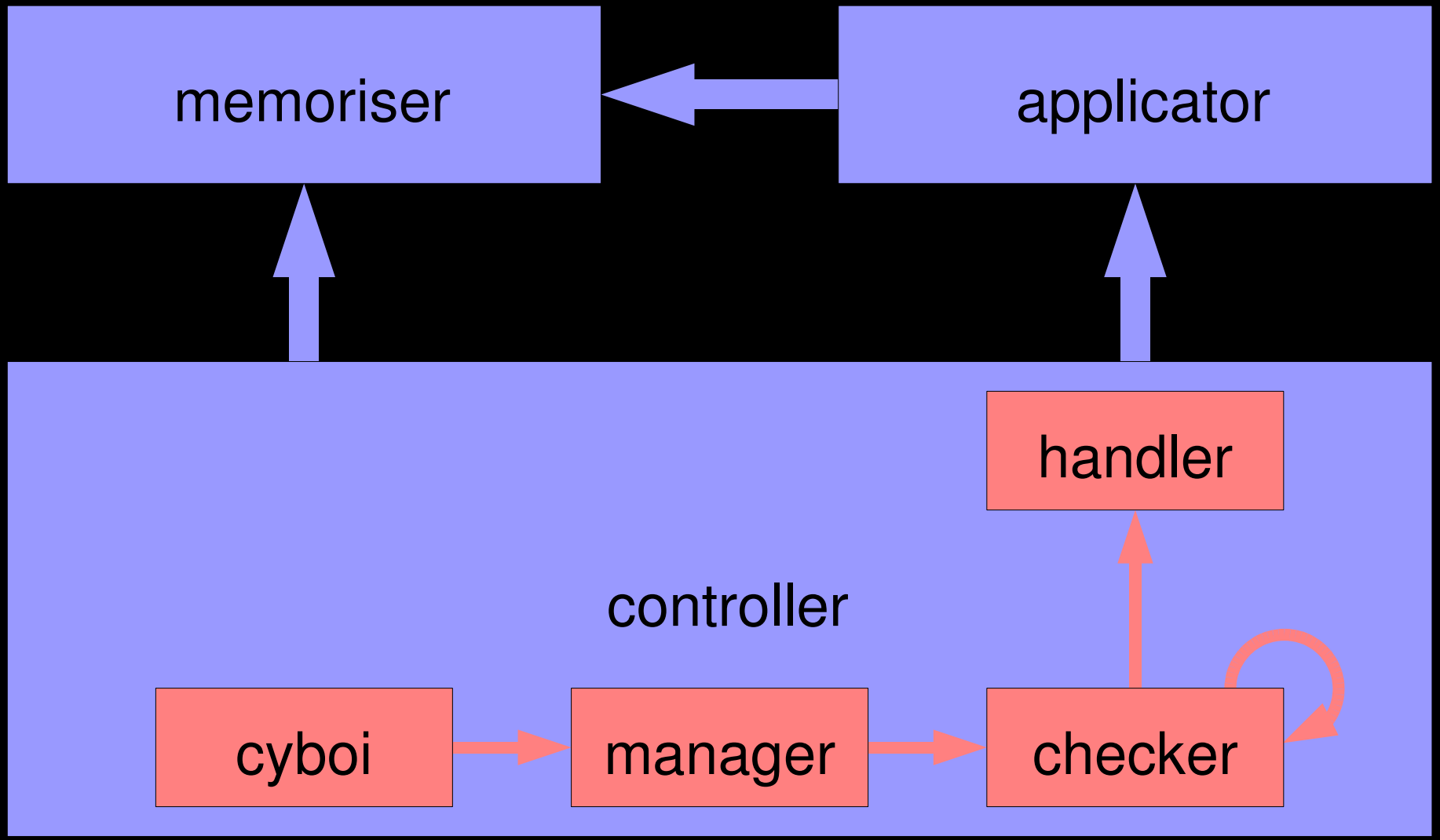
whole-part relation

meta information

template editor with double hierarchy and triple click


knowledge
cybol
control software
cyboi
operating system
hardware
computer

criterion	java world	cybop world
theory	oop (object oriented programming)	cybop (cybernetics oriented programming)
language	java	cybol (cybernetics oriented language)
interpreter	jvm (java virtual machine)	cyboi (cybernetics oriented interpreter)



cyboi part dependencies

cyboi control flow

http://www.resmedicinae.org

Res Medicinae

Revue

Residenz

Record

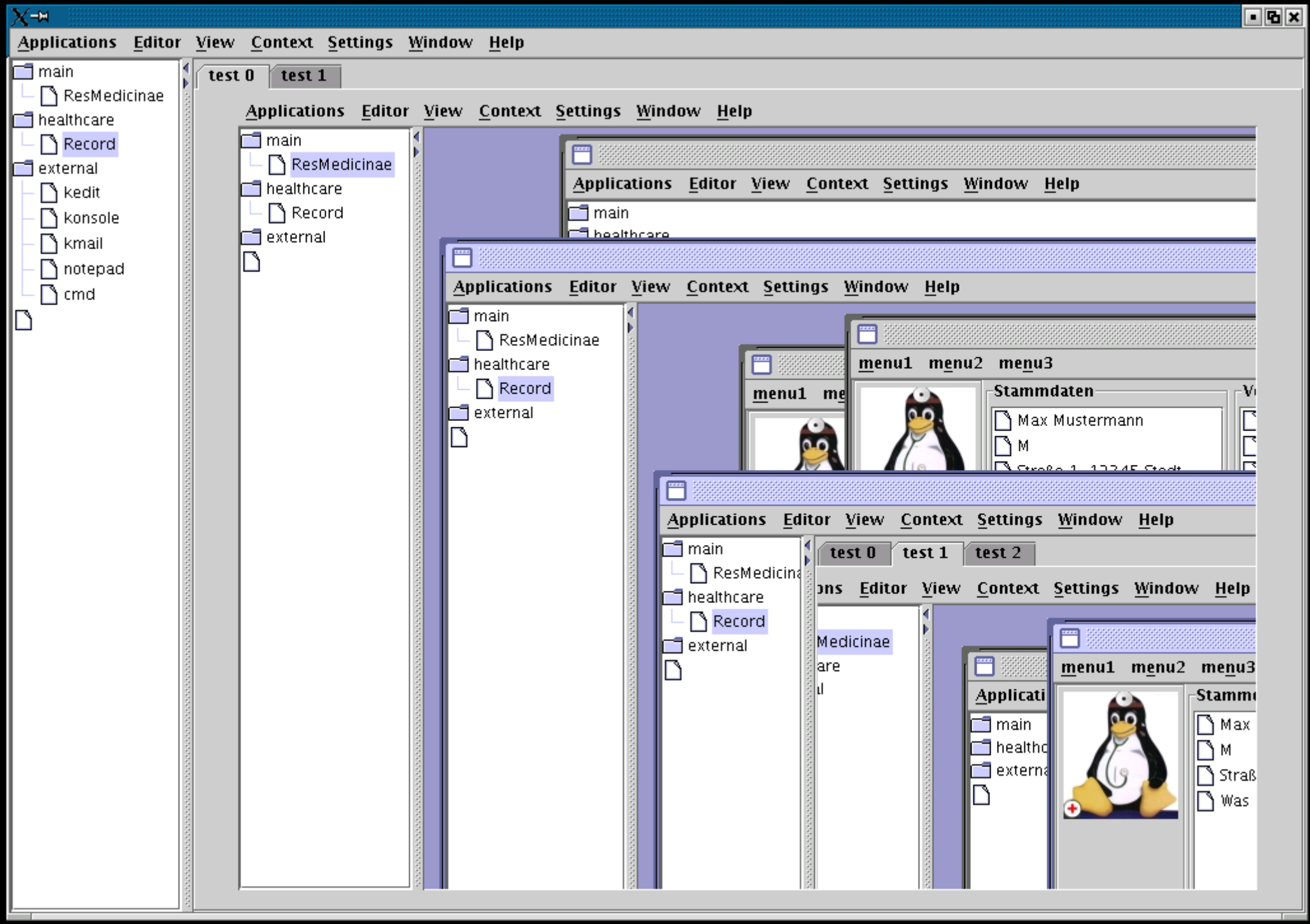
electronic health record

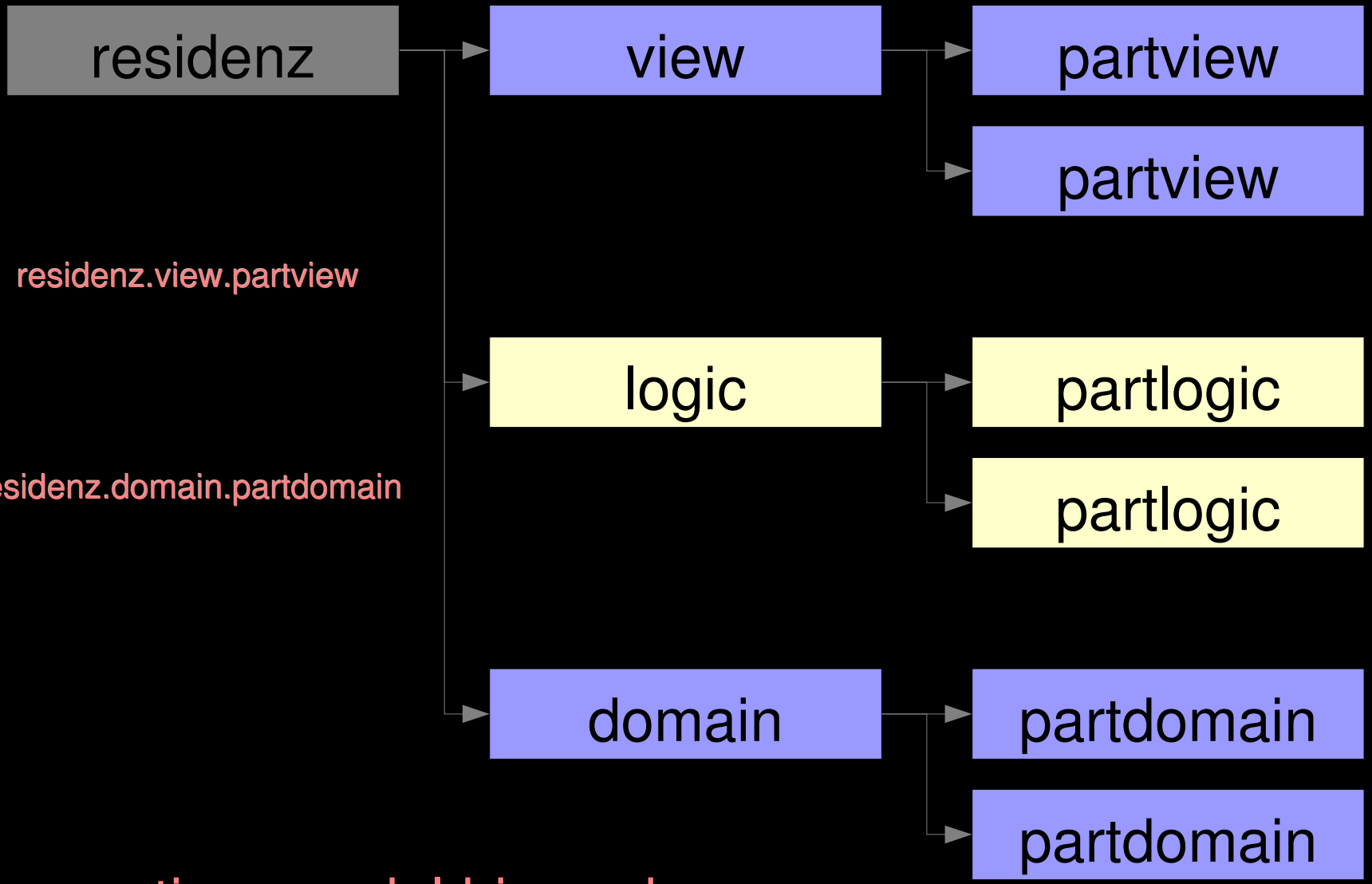
Reagenz

Roentgen

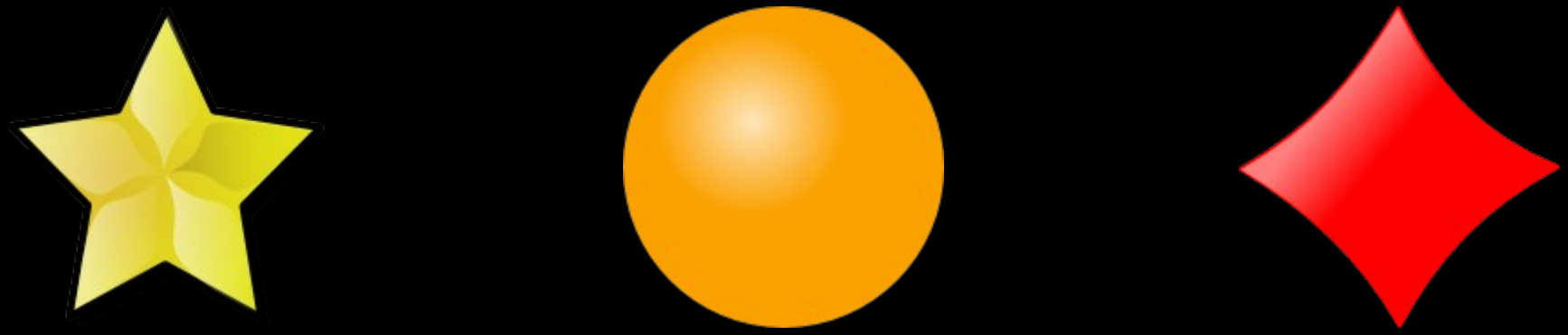
Reform

res medicinae





runtime model hierarchy



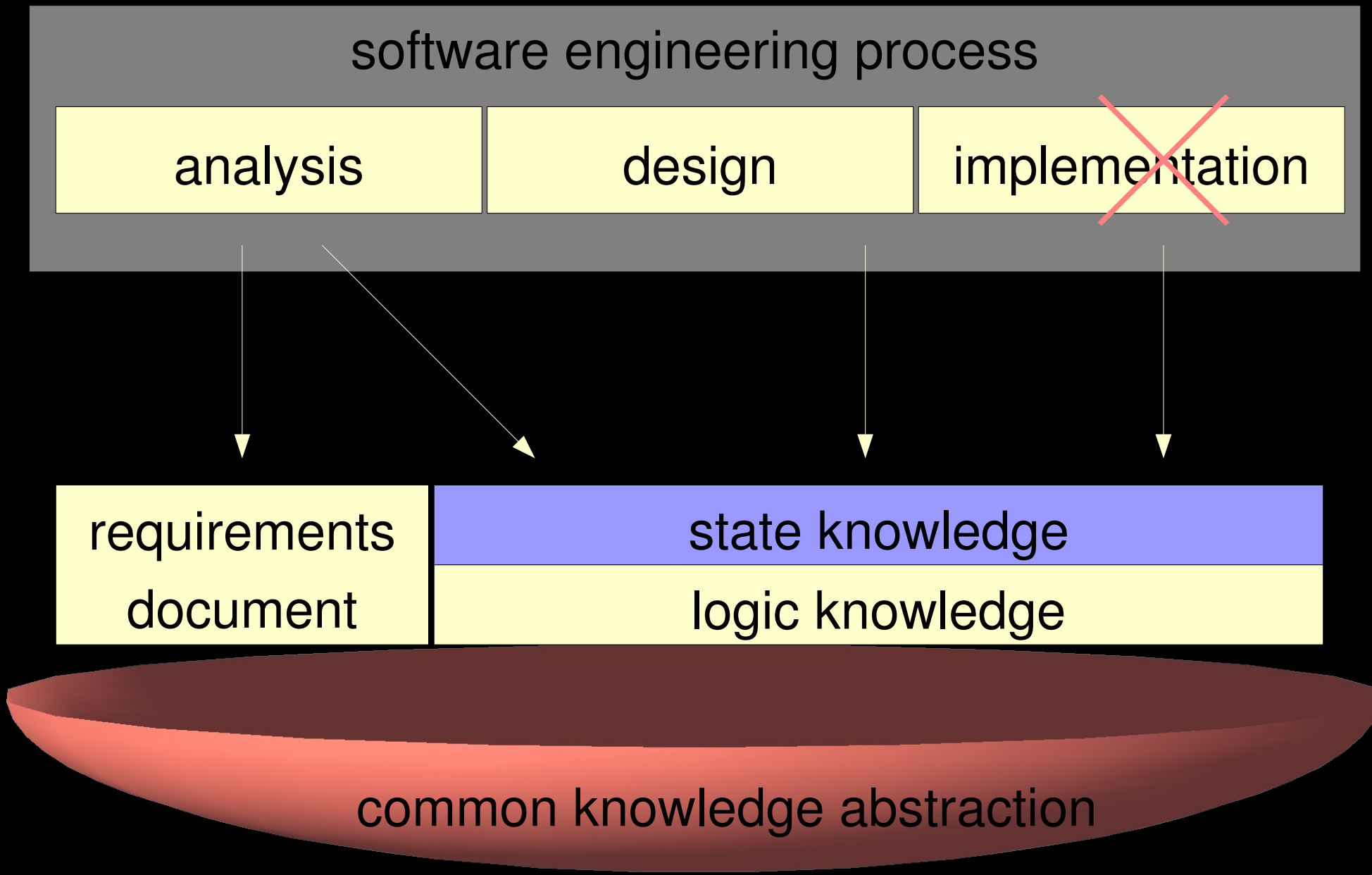
influence

instantiation

influence

template	schema	model
statics	structure	dynamics
cybol language	cybop concepts	cyboi interpreter
design time	analysis time	runtime
domain expert / application developer	knowledge architect / information scientist	systems developer

knowledge triumvirate


software engineering process
analysis
design
implementation
requirements document
state knowledge
logic knowledge
common knowledge abstraction

traditional

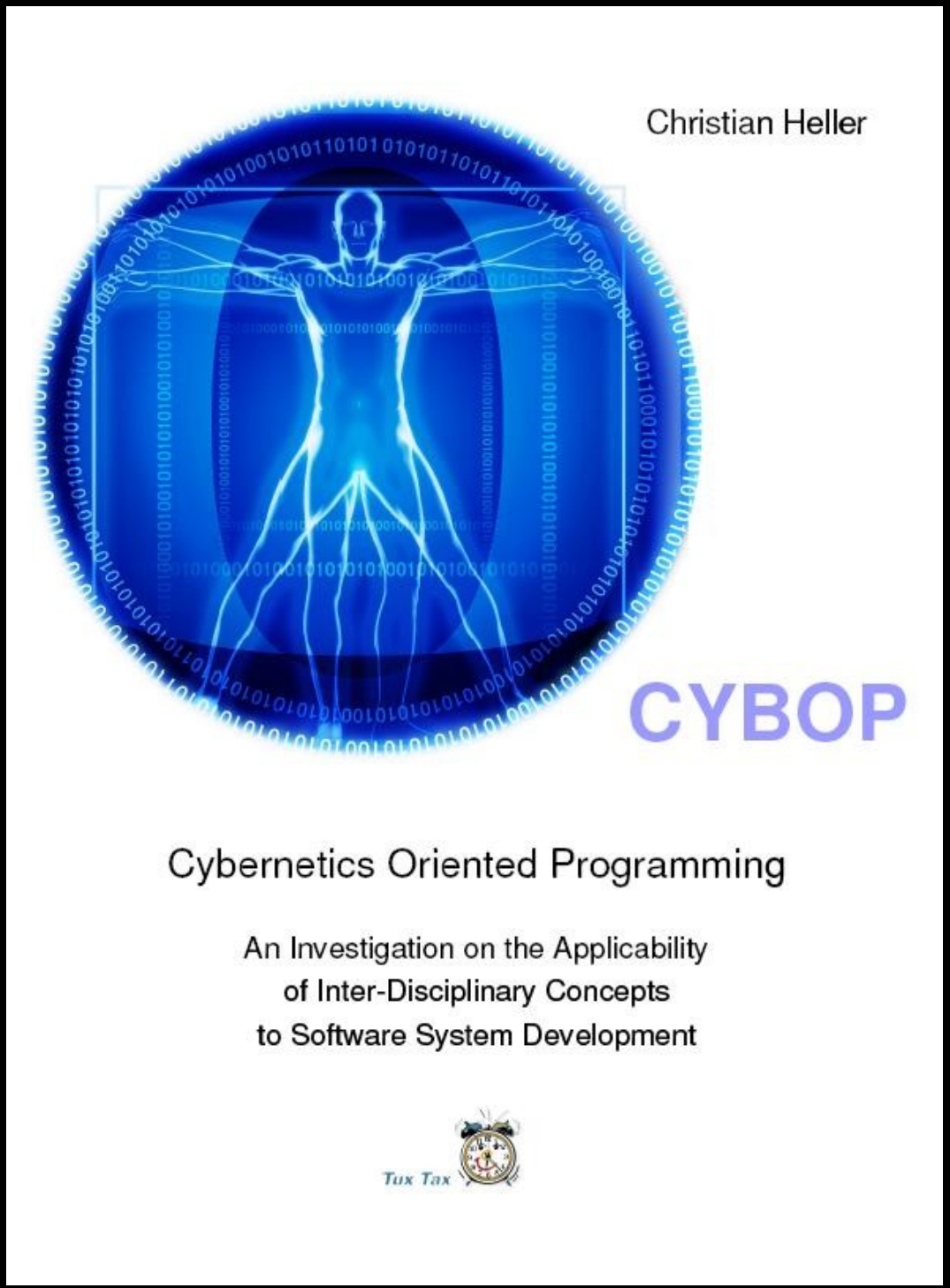
cybop

models suffer from complexity	→	one schema as memory structure
strong coupling / dependencies	→	directed acyclic graph (tree)
inflexible	→	easily extensible
difficult to maintain	→	long-life software system

limits: only standard-, no real-time applications

call for developers

- x windows, linux console, tcp sockets
- database access via sql, graphics with OpenGL/ Mesa 3D
- signalling mechanism, threading, mutexes
- port to ms windows using cygwin
- parser/ serialiser to convert different file formats
- debian package, autoconf/ automake --> official GNU
- cybol knowledge templates for various domains



thank you!

http://www.cybop.net

hardcover: 536 pages

1st edition (January 19, 2007)

language: english

license: gnu fdl

50,00 EUR