Catholijn Jonker[1], Jan Treur[1,2] , Wouter Wijngaards[1]

Reductionist and Antireductionist Perspectives on Dynamics*

[1] Vrije Universiteit Amsterdam
Department of Artificial Intelligence

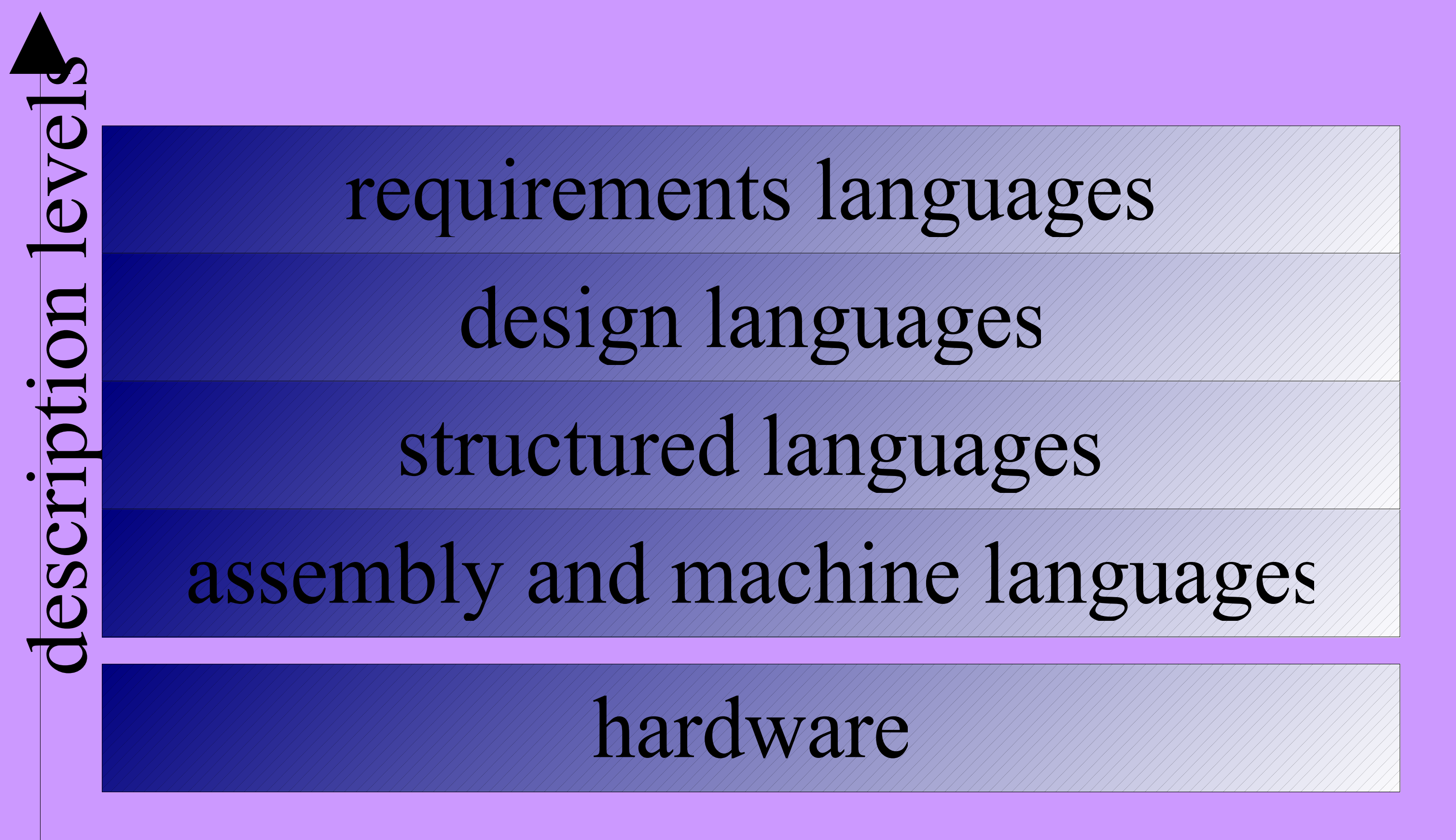
[2] Utrecht University
Department of Philosophy

*To appear in *Philosophical Psychology*, vol. 15, 2002

Overview

- The Computer Science Case
  - languages at different levels
  - reduction relations
- Higher-level Explanation
  - Jackson & Pettit, Dennett, Bickle
- The Use of Description Levels in Computer Science Practice
- Pragmatic Choices on the use of (Anti)Reduction in Practice

The Computer Science Case: Description Levels for Dynamics



Independent Semantics

- Each of these languages has its own way of referring to process instances or traces in the world, which allows empirical validation of a description.
- Moreover, this relationship between descriptions and actual process instances is formalised in the semantics of such a language.
- This picture shows that the different language levels have an independent and autonomous status; they do not depend on each other.

Multiple Realisation Relations

- Well-defined interlevel relationships between descriptions in the different languages exist
- When translating a higher-level description into a lower-level description, different translations can be made that have exactly the same effect.
- Therefore every description level is multiply realizable in lower levels.

Lowest Level: States

Machine language can be used to define state transitions, where the states are the physical states of a computer, usually represented by bit strings; for example:

10010110 00110101 11101011

Language describes transitions step-by-step.

Lowest Level: Traces

time

Accumulat or	N Z C V flags	RAM at 030B	Program Counter
00010000	0 0 0 0	00000000	00000011000000 00
00100000	0 0 0 0	00000000	00000011000000 01
00100000	0 0 0 0	00100000	00000011000001 00
01000000	0 0 0 0	00100000	00000011000001 01
10000000	0 0 0 0	00100000	00000011000001 10
10000000	0 0 0 0	00100000	00000011000001 11
10100000	0 0 0 0	00100000	00000011000010 10

Structured Programming Level: States

Values assigned to own chosen variable names:

v[2] : “Athens”

v[3] : “Paris”

v[4] : “Amsterdam”

v[5] : “London”

i : 3

j : 4

temp : “Paris”

Structured Programming Level: Traces

v[2] : "Athens"
v[3] : "Paris"
v[4] : "Amsterdam"
v[5] : "London"
i : 3
j : 4
temp : "Paris"

=============== transition

v[2] : "Athens"
v[3] : "Amsterdam"
v[4] : "Amsterdam"
v[5] : "London"
i : 3
j : 4
temp : "Paris"

Language describes sequences of transitions as a whole.

Component-based Design Language Level: Trace

time	component state
1	OE: observation_result(car_starts, neg)
2	HD: assumed(battery_empty, pos)
3	OD: predicted(lights_work), neg)
4	OE: to_be_observed(lights_work)
5	OE: observation_result(light_works, pos)
6	HE: rejected(battery_empty, pos)
7	HD: assumed(battery_empty, neg)

Language describes temporal relations between states; e.g., as in functional role descriptions

Jackson and Pettit: Program Explanation (1)

A notion of higher-level explanation, meant for special sciences such as Biology, Cognitive Science and Social Sciences.

For higher-level properties F and G,

> G occurred because F occurred

can be an adequate explanation in the following way:

> F ensures ('programs for') some lower-level property P, which causes a lower-level property Q for which G is a higher-level description.

Jackson and Pettit: Program Explanation (2)

For example, the question

Why was the vase breaking ?

can be answered by:

Because it was fragile.

Here the higher-level property of being fragile ensures or programs for the lower-level property of having a specific type of molecular structure.

Jackson and Pettit: Program Explanation (3)

'According to (Lewis, 1988), to explain something is to provide information on its causal history . . . A program explanation provides a different sort of information . . . A program account tells us what the history might have been. It gives modal information about the history, telling us for example that in any relevantly similar situation, as in the original situation itself, the fact that some atoms are decaying means that there will be a property realized - that involving the decay of such and such particular atoms - which is sufficient in the circumstances to produce radiation. In the actual world it was this, that and the other atom which decayed and led to radiation, but in possible worlds where their place is taken by other atoms, the radiation still occurs.' (Jackson and Pettit, 1990), p. 117

Dennett

'Predicting that someone will duck if you throw a brick at him is easy from the folk-psychological stance; it is and will always be intractable ***if you have to trace the*** *protons from brick to eyeball, the neurotransmitters from optic nerve to motor nerve, and so forth. '*
(Dennett, 1991), p. 42

Bickle

'Of course, the functional profiles assigned to cognitive states on Hawkin and Kandel's neurobiological account are much more fine-grained and detailed, for that account recognizes distinctions and connections that folk psychology either lumps together or leaves extremely vague . . .

Here again, however, we can expect that injection of some neurobiological details back into folk psychology would fruitfully enrich the latter, and thus allow development of a more fine-grained folk-psychological account that better matches the detailed functional profiles that neurobiology assigns to its representational states.'

(Bickle, 1998), p. 207-208

Description Languages in Computer Science

- A description at one level is multiply realizable in several possible lower-level descriptions.
- Depending on a specific context (a specific implementation environment chosen), a description at a higher level can be translated into a specific description at a lower level in an automatic manner (local reduction relation).
- Each level of description can be used to explain a particular behaviour.

Description Languages in Computer Science

- In practice, explanations are considered more valuable when expressed in terms of a higher level of description.
- Due to the fact that reductions to lower levels can be completely automated, the programmer need not have any knowledge of the lower levels.

Description Languages in Computer Science

- To increase the scope of applicability, the distance between the relevant complex physical processes and the conceptual descriptions of them has been increased.
- The success of this strategy was only possible because of the development of high-level modelling languages and supporting software environments that enabled practitioners to work on a high level of description without the need of technical expertise of the lower-level languages.

Two Perspectives on Reduction

- Reduction in a structural sense
  (i.e., a reduction relation, already established or being established, between two theories and their ontologies and laws)

- The pragmatics related to reduction
  (i.e., the use of an existing or to be achieved reduction relation in scientific practice).

Why Actually Reduce Higher-level Descriptions to Lower-level Descriptions

ontology	ontological simplification: less terms to describe phenomena in the world
assumptions	less assumptions on the world
insight	more insight in underlying lower level mechanisms for higher level phenomena
validity	empirical test possibilities at the lower level
transparency	abstraction principles have to be developed within lower-level theory
genericity	context and scope of application narrowed down to the context of the local reduction

Why Use Higher-level Descriptions in Addition to Lower-level Descriptions

ontology	additional ontologies to improve understanding
assumptions	assumptions on the world made more understandable
insight	more insight from a conceptually higher level perspective
validity	possibilities for verification and validation at different levels
transparency	analysis and design of more complex systems possible
genericity	wider scope of application

Multiple Realisation Problem (1)

Suppose P_1, P_2, … are realizers of M.

Then for each i it holds: x has P_i → x has M.

For example, C-fiber activation (P_3) implies pain (M).

However, would the converse

x has M → x has P_3

be valid ?

The answer is 'no': it may be the case that M is realised by another P_i.

So, for no i can we have a biconditional law of the form x has P_i iff x has M; no identity P_i = M.

Multiple Realisation Problem (2)

A remaining possibility is a bridge law of the form:

P_1 or P_2 or ….. iff M.

However, such an (unbounded) disjunction is not considered a legitimate property that can be expressed in the lower-level language.

Consider, for example, neurological properties of dogs and electro-magnetical properties of robots in one disjunction.

Supervenience Relations

- A notion weaker than reduction is needed.
- Higher level properties are still dependent on or determined by lower level properties.
- How ? Supervenience.
- Higher level properties supervene over lower level properties.

Sculpture Example

- Aesthetic properties of a sculpture supervene over its physical properties.
- Physically indiscernable, then aesthetically indiscernable.
- Same physical properties, then same value as art.
- Determination or dependence without reduction.

Promise of Supervenience

- A non-reductionist dependence relation, as a basis for nonreductionist physicalism.
- Consistent with both reductionism and antireductionism (neither one is implied)

Supervenience Physicalism

If something has mental property M, it has a physical property P such that necessarily if anything has P, it has M.

Local Reductions (1)

If global reduction is asked too much, then with a bit less ambition: assume context-specific reduction.

The idea is, starting from supervenience, to identify for each realizer P for M in which context it occurs as a realiser.

This context is characterised as a structure S.

Local Reductions (2)

Structure-restricted bridge laws:

If P is a realizer of mental property M in organisms or structures of type S, then the following holds as a matter of law.

If anything is a structure of type S, then
at any point in time it holds

S has M iff S has P (shorter $S \rightarrow (M \leftrightarrow P)$)

Local Reductions (3)

- Local reduction makes the multiple realisation argument against reduction without power.
- Perhaps this is all the reduction we need or could want; human psychology can be reduced to human neurophysiology. See Computer Science.
- If L is a psychological law, such a reduction would tell us how it can be derived from laws of human neurophysiology in conjunction with the restricted bridge laws.
- It would tell us how psychology is biologically implemented in human organisms.

Local Reductions (4)

- Does this decompose Cognitive Science into a number of species-specific areas ?
- The instances of M are partitioned into one group for each species. Within one group, the realizers (say P_3) have similar causal power.
- However, realizers (say P_3 and P_5) of instances in different groups will have different causal powers.
- Does this undermine that M is one scientific kind ?
- It seems that M is also decomposed into more homogeneous properties per species.

Objections ?

Multiple realizability goes deeper:

- also within one single species
- maybe even within one organism over time

However, still sufficiently similar ?

No precise and exceptionless laws in Cognitive Science can be expected ?

The success of a unitary Cognitive Science depends on how similar the neurobiology of different humans is.