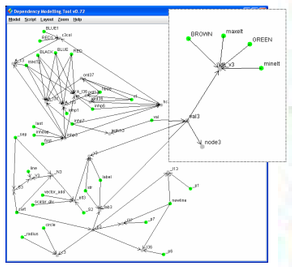
"Understanding backwards is, it must be confessed, a very frequent weakness of philosophers, both of the rationalistic and of the ordinary empiricist type.
Radical empiricism alone insists on understanding forwards also, and refuses to substitute static concepts of the understanding for the transitions in our moving life."

Understanding forwards

William James's radical empiricism is a philosophic stance centred on **the world of pure experience**. For James, knowing is rooted in **conjunctive relations** that are directly experienced.

Empirical Modelling, as illustrated in this heapsort model, creates an interactive environment framed by **observables** and **dependencies**.

The many observables associated with a particular array element are linked via the acyclic net of dependencies depicted below:

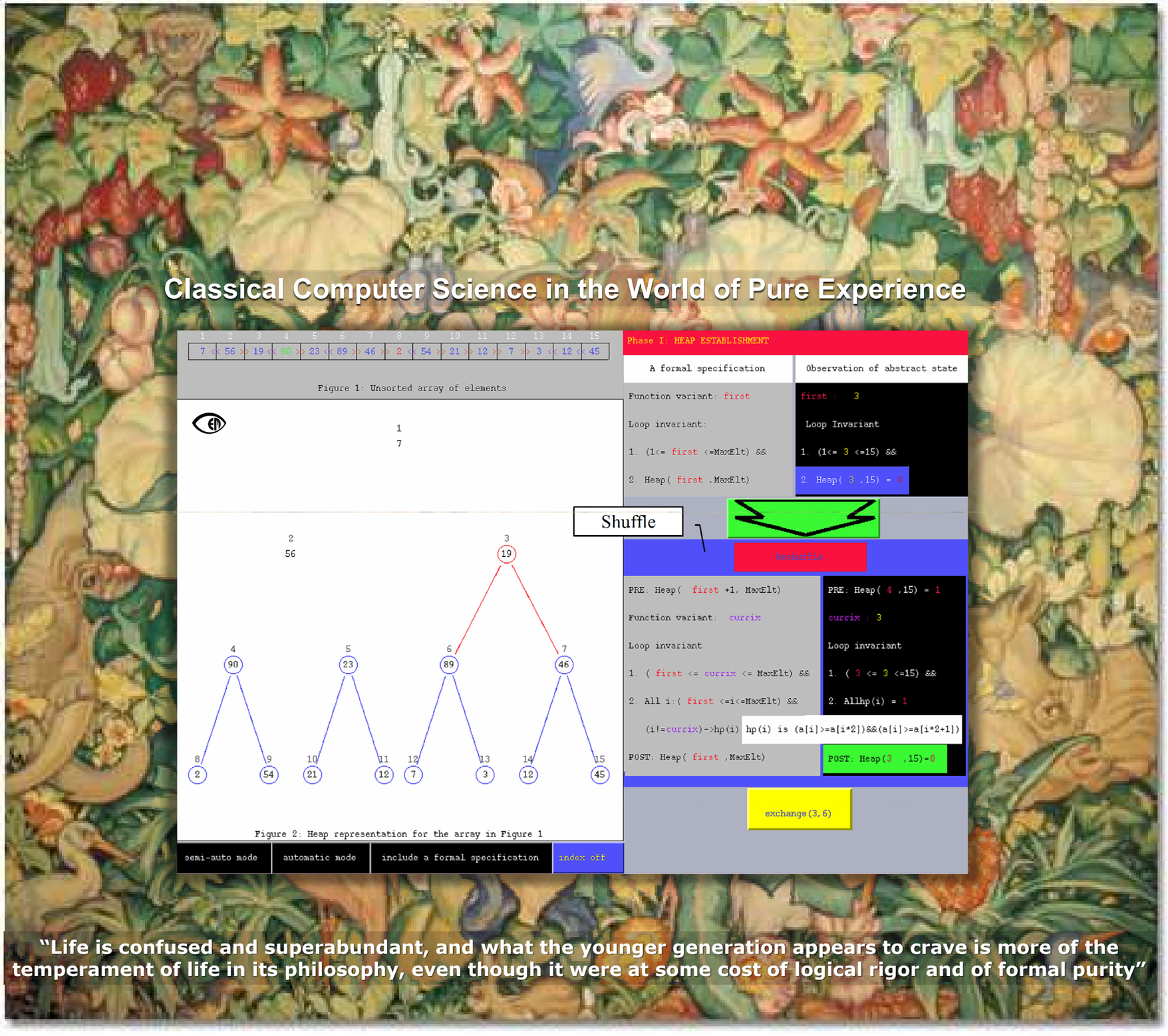


The nodes in the inset at the top right express the dependency that accounts for the colours of the array elements: the largest is displayed in brown, the smallest in green, the others in blue.

The net of dependencies maps out observables associated with a specific array index, perceived by an expert on heapsort as linked by conjunctive relations. These link an index with a location in the array and in the heap, with neighbouring nodes and logical conditions that pertain in relation to them, with colour conventions to display their key characteristics etc. They also connect the current situation with a phase in the heapsort process, and a rubric for Figure 1 that declares whether the array has been sorted.

The state sustained by this net is a playground for open interaction and interpretation by many agents. In the heapsort model, agents at the nodes of the heap can monitor and impose the heap conditions. Reconfiguring dependencies and agents transforms the viewer's experience so as to freely enhance and subvert meanings. Possible transformations might include: suppressing colour, improving the aesthetics of the heap diagram, or perturbing the binary tree structure.

Suitable transformations, organised and automated to a specific pattern, can realise a **construal** of heapsort.



Understanding backwards

… is to perform on all conjunctive relations "the usual rationalistic act of substitution – [taking] them not as they are given in their first intention, as parts constitutive of experience's living flow, but only as they appear in retrospect, each fixed as a determinate object of conception, static, therefore, and contained within itself".

Abstraction is the archetypal activity in understanding backwards. Classical programming relies essentially on abstraction - of precisely specified functionalities from an application domain, of mechanisms from the physical machine, and of systematic interaction and interpretation in use.

A formal specification identifies heapsort as a conceptual object, independent of a specific implementation. It expresses transitions only in a contrived and implicit way, making no reference to states and state changes currently being experienced.

The abstract context for interpreting a program is so critical that a bug in the machine, flaw in the environment, or lapse in the user protocol typically explodes meaning. In contrast, every interaction with the construal invites interpretation, in whatsoever way it transforms the net of dependencies.

In classical computing, the invariants of a program impose the constraints needed to maintain its integrity. Thus viewed, they serve as 'abstract concepts' of which James writes: "Every abstract concept as such excludes what it doesn't include".

In the heapsort model, features such as the current phase in the heapsort process are identified as observables that can be discerned only when certain sophisticated patterns of engagement with the construal are respected. The invariants in the specification are also of this nature.

The classical account of heapsort as an efficient and oblivious rule-based activity is well-matched to scientific explanations that eschew overt agency. The heapsort construal, in contrast, supports variants of the sorting activity that admit the possibility of random interventions, such as change the values of keys during the sorting, and can exploit sophisticated agency based on observing the invariants and the phases to resume the sorting process.

Heapsort as percept

"... subjectivity and objectivity are affairs not of what an experience is aboriginally made of, but of its classification"

" … as reality is created temporally day by day, concepts ... can never fitly supersede perception"

Heapsort as concept