Titles and abstracts of the HEPIC séminaire session
Programs — technological artefacts?
April 29, 2021, 9h30-13h
Virtual meeting
https://calcul.hypotheses.org/

Speaker: Henri Salha
Title: The two sides of design and implementation

Abstract: Design is a crucial part of any programming project but has not been yet a major topic of investigation for philosophers. It shows however a dual prescriptive / descriptive character that has puzzled computers scientists for a long time, which is not common in other technological artefacts. This paper reviews main interpretations provided on the topic and offers one based on the fact that programming designs describe a target state of affairs – therefore prescribing a behavior not only to the program, but also to its users and its environment. This analysis of design will allow us to show a remarkable result: that specifications have actually a two-sided normativity. They do not only give norms to the program itself, but also to its context of use (e.g. how users must behave for the program to work properly).
This also implies that programming projects have also a two-sided implementation: not only must the program be installed on hardware, but end-user processes must also be often be adapted, trainings provided, rules of interaction redefined, etc. We will link this result to a general statement about the normative status of all technological artefacts.

Speaker: Ray Turner
Title: Computational intention

Abstract: The core entities of computer science include formal languages, specifications, models, programs, implementations, semantic theories, type inference systems, abstract and physical machines. While there are conceptual questions concerning their nature, and in particular ontological ones, our main focus here will be on the relationships between them. These relationships have an extensional aspect that articulates the propositional connection between

the two entities, and an intentional one that fixes the direction of governance. Two core intentions involve “representation” and “abstraction”.

An analysis of these two aspects (extension and intention) will drive our investigation; an investigation that will touch upon some of the central concerns of the philosophy of computer science.

Speakers: Maarten Franssen and Sjoerd Zwart
Title: Ray Turner’s conception of computer programs as technical artefacts – a critical assessment

Abstract: Many authors present computer programs as enigmatic entities because there appears to be an ‘abstract side’ as well as a ‘physical side’ to them. Ray Turner has attempted to resolve this enigmatic aspect, or ambiguity, by interpreting it as a duality and by connecting it to the dual-nature view of technical artefacts, which results in his thesis that computational artefacts are the technical artefacts of computer science.
Our aim in this talk is first of all to discover what Turner’s position amounts to.
In doing so, we criticize him on

(a) his interpretation of the dual-nature view of technical artefacts, in particular his interpretation of the notions of functional and structural descriptions,
(b) his mapping of the abstract/physical duality of computer programs on the intentional/ physical duality of technical artefacts, and
(c) the lack of clarity of his view of computational artefacts and in particular of the central notion of (computer) program.
With respect to point (c), Turner mentions programs as just one in an implausibly large range of entities (occasionally) classified by him as technical artefacts – program languages and program implementations or executions are so termed as well – but just as well includes programs in an equally implausibly large range of entities classified by him as mathematical objects – algorithms and specifications or descriptions are so termed as well – even if programs are stated to be not 'purely' mathematical objects.
Secondly, by way of analyzing Turner's position, we aim to arrive at a re-evaluation and re-formulation of what Turner refers to as 'the ontology of computer programs'. We compare it with and contrast it to similar oppositions already well-known and well-discussed elsewhere, e.g. in language, art and science.