

Abstract State Machine

Notation : ASM

Description : Abstract state machines support the practitioner in exploiting their power of *abstraction in terms of an operational system view* which faithfully reflects the natural intuition of system behavior. Even the knowledge-base of experts has an operational character and guarded command form: “in *this* situation, do *that*”. A solid model of automata which are amenable to mathematical and experimental analysis. This is further enhanced by the ability to link requirements capture to detailed design and coding. Finally, it provides on-the-fly documentation which can be used for inspection, reuse, and maintenance.

By combining abstraction and stepwise refinement and putting them on a semantical, rather than a syntactical basis, and combining them with the operational nature of machines, a pseudocode can be made modularized, refined, and granular, in a logically relates the sensible parts for the viewer. What this means is assigning operands and operators to specific, discrete functions, without conflating either the nature of their being, or of their operation. When the parts of multiple functions are unified behind a single set of defined concepts, the possibility for confusion is massively reduced, because the differences and similarities between each function present themselves to the user rather than having to be painstakingly weeded out, through trial and error, by the user.¹

1. E. Borger and Stark Robert F. 2003. *Abstract state machines: a method for high-level system design and analysis*. Springer.