

Java virtual machine started as a single product that was supposed to be executed in browser, but then was split into different standards depending on the hardware capabilities:

- Java Standard Edition
- Java Enterprise Edition
- JavaCard (smartcards)
- J2ME (phones/embedded devices)

I think if EVM wants to keep its lead (currently it does have lots of market momentum) there has to be a process to define different EVM editions depending on the capabilities of the ledger, taking into account that many EVM implementations will not run on Ethereum blockchain.

For example EVM that runs on Ethereum does not include floating point numbers, but if other EVM implementations may want them, then there should be a profile supporting floating point numbers.

The question is how this process would work ... For example when Sun Microsystems introduced Java, they introduced a formal Java Community Process where all industry members would participate with voting, reviews, committees etc ...