**Virtual Machine**

**Abstract:**

The complete instruction-by-instruction simulation of one computer system on a different system is a well-known computing technique. It is often used for software development when a hardware base is being altered. For example, if a programmer is developing software for some new special purpose (e.g., aerospace) computer X which is under construction and as yet unavailable, he will likely begin by writing a simulator for that computer on some available general-purpose machine G. The simulator will provide a detailed simulation of the special-purpose environment X, including its processor, memory, and I/O devices. Except for possible timing dependencies, programs which run on the “simulated machine X” can later run on the “real machine X” (when it is finally built and checked out) with identical effect. The programs running on X can be arbitrary — including code to exercise simulated I/O devices, move data and instructions anywhere in simulated memory, or execute any instruction of the simulated machine. The simulator provides a layer of software filtering which protects the resources of the machine G from being misused by programs on X.