

the same UseCase are categorized so that the intensity and temporal properties of the UserRequests generated by each **UserCategory** can be described uniquely with a **RequestPattern**.

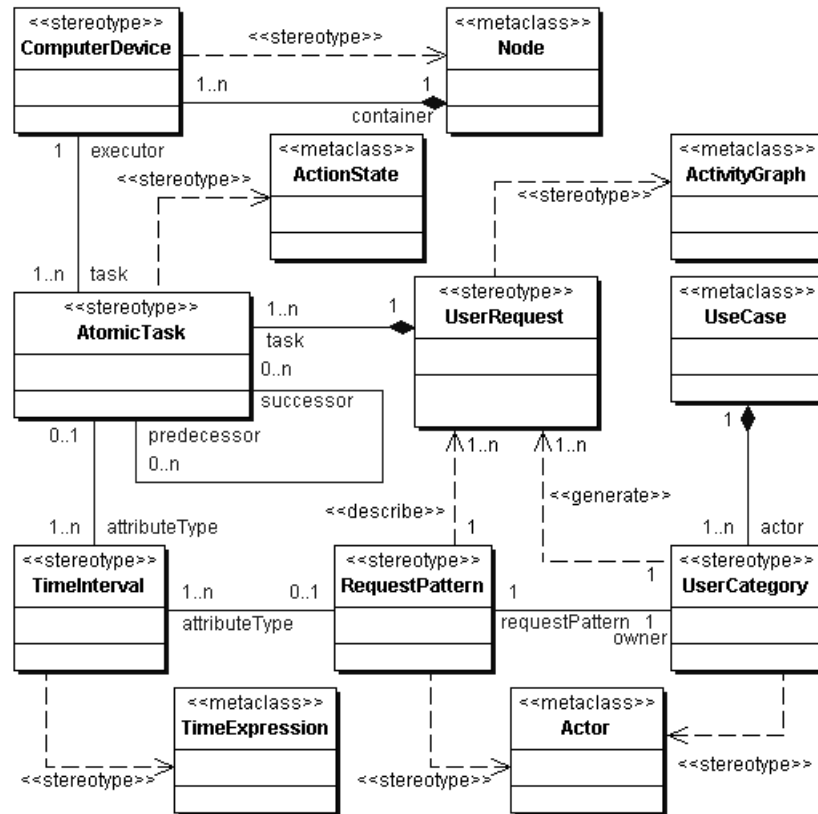


Figure 2: UML extensions in the software performance prediction UML profile

The processing of a UserRequest is abstracted as a sequence of **AtomicTasks**. The AtomicTasks are related to each other in a predecessor/successor relationship. Each AtomicTask is not decomposable and its work can not be interrupted. An AtomicTask is executed solely by one physical **ComputerDevice**. A ComputerDevice may execute more than one AtomicTasks. The ComputerDevices are contained in the runtime computational **Nodes**. Temporal properties of AtomicTasks and RequestPatterns are specified by using **TimeIntervals**.

UML extensions stated above have been used to extend functional software models, primarily use case- design- and deployment models, with information about 1) user request pattern and 2) which computer devices are required for executing the system functionality, and how their parameters and quantitative behavior are. Based on the extended software models, queuing network models are built for software performance evaluation.