ABSTRACT

The prime objective of this work is to establish a methodology applied in a practical visual modeling tool—ProcessModeler (PM), for design, verification and simulation of process definitions implemented in a workflow management system (WfMS).

With PM, a process definition, which is a network of activities and connected by links, can be easily modeled as a graph. In accordance with the process definition, business processes can then be automatically created at one of the start activities and be routed along the links from activity to activity in a WfMS. The diverse routing options of links allow the activity execution thread of a business process to be dynamically determined according to the real world circumstances.

The activity network of a process definition can be simply and flexibly designed with PM and can, thus, be very complex. However all the activities within a process definition must be reachable from a start activity. No infinite cycle in a process definition is allowed. According to the network structure, PM determines for the workflow engine join activities and other workflow control data of a process definition. At a join activity, parallel activity execution threads will be merged into one thread. Work items waiting for joining at the join activities may yield at run-time a deadlock situation of a business process. Therefore, PM detects the deadlock cycles of a process definition and assigns join priorities to the join activities so that the workflow engine can release deadlock situations.

Uneven parallel activity execution threads in a process definition can prolong the duration of a business process. Shortage or unbalanced allocation of human and material resources demanded for the execution of the activities can also lengthen the duration. The simulator integrated in PM represents sufficiently cause-and-effect relationships of a real world WfMS and can thus, be used to make experiments of the designed process definitions. The simulation reports offer insight to the workloads, bottlenecks, resource allocation, throughput, productivity, and overall business cycle. By analyzing these, immediate decisions can be made to alter a process definition by reallocating resources, changing activity network, eliminating redundancy, or altering priorities of work.

To simulate a WfMS, a stochastic and dynamic system, a broad body of input data should be estimated upon the collected data or guessed by specialists so that they represent the features of the real world system as well as possible. The validity of the results obtained from a simulation study is influenced by such factors as the techniques used in the collection of data and the analysis methods used in summarizing the data. Further, prior to its use, the simulator should be validated, or shown to actually represent the system being studied. Therefore, the theory and methods requisite for the proper development and operation of the simulator are presented in this work.

Some algorithms for process definition and especially simulation are built upon assumptions. The assumptions specify prerequisites, constraints, or principles for process design and simulation. They should not diverge from the behavior of the real world business processes.

The algorithms in this work are described for a general-purpose programming language, such as C++ and Visual Basic. They provide deep insights into the actual logical intricacies of PM.