VIPtool

A partial order based simulation tool for high-level Petri nets

Jörg Desel, Thomas Freytag, Institut AIFB, University of Karlsruhe, Germany
Andreas Oberweis, Torsten Zimmer, Institut für Wirtschaftsinformatik, University of Frankfurt, Germany
desel | freytag @aifb.uni-karlsruhe.de
oberweis | zimmer @wiwi.uni-frankfurt.de

Short description

VIPtool is a tool for editing, simulation and validation of high-level Petri nets. It implements concepts that are in the focus of a project named ***VIP*** (***V***erification of ***I***nformation systems by evaluating partially-ordered runs of ***P***etri nets). This project is a cooperative work of researchers at the universities of Frankfurt and Karlsruhe and is sponsored by the German Research Society (DFG). The main concern of this project is to establish a simulation and validation concept for Petri nets that is not based on the generation and evaluation of totally ordered sequences of transitions but on partially ordered execution runs (process nets).

Theoretical background

There are two general views at the dynamic behaviour of a net model: The first one - called the *sequential semantics* - looks at the set of *firing sequences* of a net. The second one - called the *causal semantics* - looks at the set of partially ordered runs (or *processes*) of a net. Whereas the causal semantics is favorized in Petri net theory because of its ability to handle concurrency, the area of applications still is dominated by the sequential semantics because of their easy and straight-away definition.

There are also two ways to check (desired or undesired) properties of a net model: The first one is to apply *analysis methods* from net theory deciding a property essentially by analysis of the state space. This very clean way becomes inapplicable when systems are large and their state space suffers from the *state explosion.* The second approach is not to consider all possible executions of a net but only a well-chosen subset determined by *simulation.* The given property is then checked against 'empirical' data - comparable to the systematic testing of a program.

The ***VIP*** project tries to find a way betwee these approaches, preserving the advantages and avoiding the disadvantages of each:

- simulate systems modelled by high-level Petri nets by generating partial ordered runs,
- check given system properties by evaluating these runs,
- specify these properties in a user-oriented graphical way and
- visualize dynamic system behaviour by means of its causal semantics
- optimize system performance through investigation of causal behaviour

Implementation

VIPtool is mainly written in *Python*, a freely available, object-oriented and interpreted script language. Runtime-critical algorithmic parts (such as the simulator and the verificator) are written in *ANSI-C* and are compiled and linked directly into the interpreter as dynamic link libraries. The graphical user interface is based on an built-in extension of *Python* that gives full access to the *Tcl/Tk* functionality.

Hardware requirements

VIPtool is available for *Intel-PCs* running *Linux* and *Sun* workstations running *SunOS*. A port to *Windows 95/Windows NT* is planned for the future.

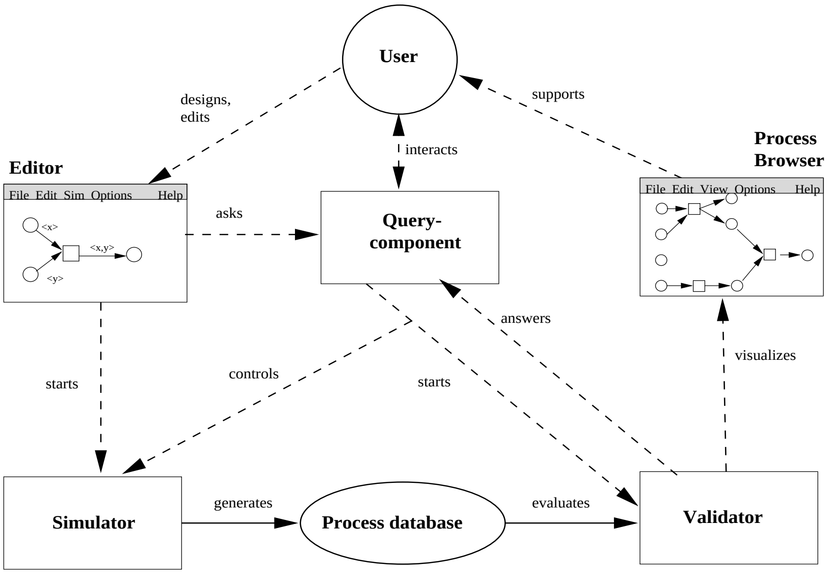
Software requirements

VIPtool requires *Python* version 1.4, *Tcl* version 7.5, *Tk* version 4.1 and an ANSI-compatible C compiler for building the sources.

Structure and functionality

VIPtool consists of six main components:

- A net *editor* where nets can be edited and system properties can be graphically specified
- A *process database* containing all simulated processes
- A *query component* which accesses the process database and controls the property-based part of the simulation policy
- A *verificator* that does the 'pattern matching' in order to validate the given property against the contents of the process database
- A *simulator* which generates processes of the net according to a dynamic control mechanism which includes global termination rules, conflict resolution rules, firing probabilities and rules that influence the simulation policy depending on the specified property
- A *process browser* that visualizes process nets and gives additional information about the validation of the property



Contact

For further information about ***VIPtool*** please contact
Thomas Freytag, Institut AIFB, Universität Karlsruhe, D-76128 Karlsruhe
e-mail freytag@aifb.uni-karlsruhe.de - phone ++49-721-608-4063 - fax ++49-721-693717.