

A Master Clock for PAGODA  
Andy Poggio  
7 Jun 05

*Does anybody really know what time it is?  
Does anybody really care?*

*--Chicago Transit Authority*

In order for our radio network simulation PAGODA to change its behavior and move through different states, e.g. operation phases or node failure modes, it needs to have a master clock which advances during PAGODA runs.

The functions of the master clock that must be implemented are:

**Allowing components to indicate their interest in the time**

The master clock component will be initialized with a list of components that want to be informed of clock ticks.

**Advancing the clock -- ticking**

Clock ticking should be a relatively high frequency occurrence in PAGODA. To insure such ticking, the clock tick rule is set to fire without restriction once the clock is told to start ticking. By using fair rewriting (frewrite), rather than ordinary rewriting (rewrite), the clock ticks at an appropriate rate. Only frewrite should be used with PAGODA; rewrite allows for only very infrequent ticks.

**Firing component rules dependent on the clock**

Any rules that depend on the clock must be fired in a timely fashion (so to speak). Upon each tick, the master clock component will walk through the list of interested components, sending a tick msg to each.

**Allowing desirable asynchrony in the system**

The clock should be, and is, able to tick out of sync with other state changes. Note that any individual component may have its own clock that is synchronized to, but runs at a different speed than, the master clock.