



Universiteit
Leiden
The Netherlands

Time at your service : schedulability analysis of real-time and distributed services

Jaghooori, M.M.

Citation

Jaghooori, M. M. (2010, December 20). *Time at your service : schedulability analysis of real-time and distributed services*. Retrieved from <https://hdl.handle.net/1887/16260>

Version: Not Applicable (or Unknown)

License: [Leiden University Non-exclusive license](#)

Downloaded from: <https://hdl.handle.net/1887/16260>

Note: To cite this publication please use the final published version (if applicable).

Propositions belonging to the PhD dissertation
Time at Your Service: Schedulability Analysis of Real-Time and
Distributed Services
by Mohammad Mahdi Jaghoori, defense scheduled on 20 December 2010

1. The correct functionality of a real-time system depends on the correct choice of the scheduling strategy; then, one may improve the quality of service by improving on the scheduling strategy (this thesis).
2. Multi-threading paradigm, as in Java, brings too much interdependency to threads as the units of concurrency; asynchronous actors are, on the contrary, a natural fit for distributed deployment (this thesis).
3. A good schedule brings asynchronous actors to concurrence.
Scheduling of tasks is traditionally deferred to the operating system, which by its nature is not aware of particular needs of specific applications; assigning application-specific schedulers to actors improves the overall quality of service (this thesis).
4. Besides observable actions, the refinement relation between real-time actors and their behavioral interfaces may observe deadlocks as well as passage or stoppage of time (this thesis).
5. Object orientation is a good basis for compositional analysis. To analyze an object in isolation, we use its behavioral interface as a driver; the object behavior is driven by supplying the inputs that are allowed in its behavioral interface (this thesis).
6. Schedules are made to be broken; deadlines are set to be missed.
The thesis at hand is an attempt to remedy this.
7. Thou shalt use the rightful channel for communication.
8. To pioneer, a software engineer needs a taste of philosophy.
9. A philosopher might better arrive late than never (see Chapter 5.3 of this thesis).
10. Schedulability analysis is as hard as its pronunciation.
11. The solution to “I don’t have time” is “use a better scheduler”.