E. Douglas Jensen's
Real-Time for the Real World

Home | Search | Contact Me

My personal manifesto about the widely misunderstood field of real-time computing...

Navigation

Introduction
About Me
Real-Time
Distributed Real-Time
Distrib. Real-Time Java
Real-Time Java
Real-Time CORBA
Real-Time Resources
Our Documents

Last updated: 02/29/2004 17:18:02

Deadlines

The best-known example of a time constraint is a deadline, but it is a simple one and there are many others (as we will discuss). The use of deadlines in real-time computing is a relatively recent small fraction of the overall theory and practice of deadline-based resource management in various fields (notably logistical fields).

In particular, the real-time computing community historically focuses primarily on hard deadlines (e.g., [ ]). The traditional model of an action having a *hard deadline* at time t_d is simply that the action's completion either meets or misses its deadline, as illustrated in Figure 1.

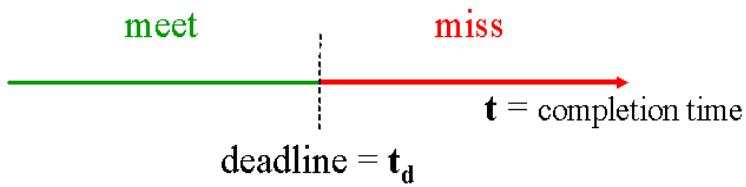

Figure 1: Traditional Hard Deadline

(The semantics of a deadline -- i.e., the specific way in which system timeliness depends on whether any particular deadline

News

Real-Time:

Real-Time Overview
Time Constraints
Deadlines
Time/Utility Functions
Time Constraints Scopes and Priorities
Sequencing
Sequencing Criteria
Timeliness Optimality
Predictability
Hard and Soft Real-Time
Sequencing Algorithms
Worked Examples
Coastal Air Defense

AWACS Tracker

History

is met, such as whether a miss constitutes a failure -- is not part of the definition of a deadline, contrary to popular misconception in the real-time computing community (e.g., [ ]). These consequences are properly specified as part of the sequencing optimality criteria.)

When the term "deadline" is used without the qualifier "hard," it refers to the general case of a deadline -- a soft deadline, of which a hard deadline is a special case: the action is either more or less timely, depending on what its completion time is with respect to its deadline $\mathbf{t_d}$ [Pinedo 02].

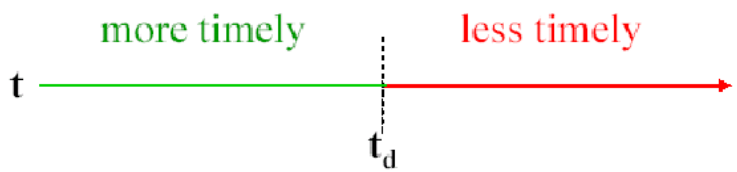

Figure 2: Traditional Deadline

(The term "soft deadline" is almost always used incorrectly in the real-time computing community, as explained on the sequencing page.)

"More or less" timely with respect to a deadline is always measured in terms of "lateness" and its derivative cases "tardiness" and (to a lesser extent) "earliness:"

- *lateness* = completion time - deadline
- *tardiness* = max[0,lateness]
- *earliness* = max[-lateness,0]

as illustrated in Figure 3.

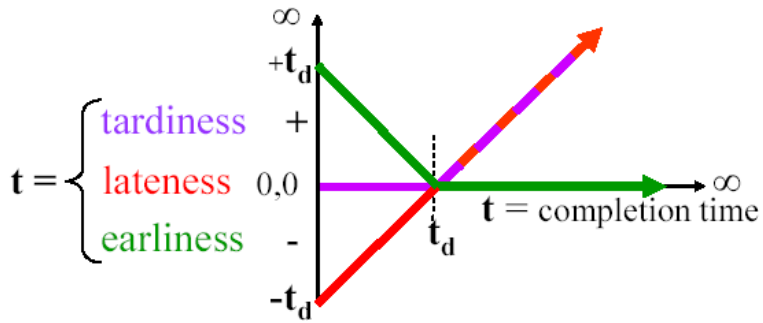

Figure 3: Deadline Timeliness Metrics

The best (most timely) case with respect to a deadline is that the action completes in zero time, so

- lateness = completion time - deadline = 0 - $\mathbf{t_d}$ = -$\mathbf{t_d}$
- tardiness = max[0,lateness] = max[0, $\mathbf{t_d}$] = 0
- earliness = max[-lateness,0] = max[$\mathbf{t_d}$,0] = $\mathbf{t_d}$

The case where the action completes at its deadline results in

- lateness = completion time - deadline = $\mathbf{t_d}$ - $\mathbf{t_d}$ = 0
- tardiness = max[0,lateness] = max[0,0] = 0
- earliness = max[-lateness,0] = max[0,0] = 0

An example case where the action completes at twice its deadline results in

- lateness = completion time - deadline = **2*$\mathbf{t_d}$** - $\mathbf{t_d}$ = $\mathbf{t_d}$
- tardiness = max[0,lateness] = max[0, $\mathbf{t_d}$] = $\mathbf{t_d}$
- earliness = max[-lateness,0] = max[**-$\mathbf{t_d}$**,0] = 0

On the next page, we show that deadlines are a special, limited, case of a more general and expressive model of time constraints: time/utility functions.

References

Pinedo 02 Pinedo, M., *Scheduling: Theory, Algorithms, and Systems*, 2nd edition, Prentice Hall, 2002, ISBN 0-130-28138-7.

Next: Time/Utility Functions

Back to Time Constraints

Add to Favorites

Print Page

Download a PDF copy of this page

View Site Changes XML | **Site Updated 02/29/2004** | **Legal**