




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 17:48:40

 Outline
--

Real-Time

Motivation and Introduction

This page and its subsequent linked pages (see the Outline in the right-hand column) discuss the fundamental concepts and terms of "real-time" computing from my point of view.

The motivation for this discussion is to improve the precision and completeness of discourse in both the practitioner and researcher communities, especially with regard to general (i.e., soft) real-time. That will help elevate the practice of real-time computing from an ad hoc craft to a engineering and scientific discipline, which in turn will result in better and more cost-effective real-time systems.

A summary of my motivation for this manifesto is as follows.

- Traditional real-time concepts and techniques are focused primarily on the niche of relatively simple, static, centralized, device-level, autonomic monitoring and control subsystems. But, contrary to popular misconception, many of the most important and technically challenging real-time computing systems are outside this niche. The instances of particular interest to me are in the field of military combat systems (e.g.,

Real-Time:

Motivation and Intro

[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](#)

[[DoD Network Centric Warfare](#)]), but prominent civilian fields include industrial automation (e.g., [Echelon 4](#)) and telecommunications.

[Coastal Air Defense](#)

[AWACS Tracker](#)

[History](#)

- Even in the classical real-time niche, all of the most fundamental concepts are misunderstood, over-simplified, ill-defined, and contradictory.
- Practitioners (users, vendors) often take recourse in Justice Potter Stewart's statement about pornography – "I can't define it ("hard," "soft," "predictable," "real-time," etc.) but I know it when I see it." This makes the practice of real-time computing most commonly be a craft (or worse – ad hoc tinkering and tuning) instead of science and engineering (like alchemy vs. chemistry). Imagine if electrical engineers had correspondingly vague perceptions and disagreements about their most fundamental concepts, like voltage, current, resistance, etc.
- The real-time research community historically has been of only very limited assistance with this problem. They have consensus on a precise technical (and correct) definition of "hard real-time," but left "soft real-time" to be tautologically defined as "not hard" – that is accurate and precise, but no more useful than dichotomizing all colors into "black" and "not black."



- In the technically accurate sense used by researchers, hard real-time is an even smaller niche than the traditional real-time niche. There are two reasons for their narrow focus.
 - First, theory for hard real-time computing is drastically easier than theory for soft real-time computing (I have long observed that "Hard real-time is hard, but soft real-time is harder") - e.g., compare deterministic scheduling theory with stochastic scheduling theory.
 - Second, real-time is unique in computer science with respect to its paucity of opportunities for researchers to be realistic experimental users of their own work. Researchers in all other aspects of computer science can apply their research results realistically in their own labs. But real-time researchers rarely have access to non-trivial real-time systems for industrial automation, defense, telecom, etc. Electric model railroads and such do not enable someone to perceive and appreciate what the important problems are in enterprises at all levels "from sensor to CEO" (or "Sensor to Commander" in the DoD domain).
- Consequently, traditional concepts and techniques, and traditional real-time theories, limit the kinds and capabilities of real-time systems that *can* be built, and raise the costs of many of those non-trivial ones that *are* built. They do not scale up to the general case of larger, more complex, more dynamic, more adaptive, more distributed, higher level, higher order control, real-time computing systems.
- Overcoming these limitations requires a new, more general and expressive, real-time paradigm – one which applies equally well to the whole continuum of real-time or more generally *time-critical* systems (analogous to all the colors which are "not black" as well as to "black"). Such a paradigm must provide precise concepts and

definitions for such terms as "hard," "soft," "predictable," "deterministic," and of course "real-time." Despite the fundamental role that "predictability" plays in real-time computing, nowhere in the real-time research literature (not to mention the practitioner documents) is it defined (much less defined formally or even precisely).

- The manifesto on this site offers an empirically successful contribution to that end. I was one of the very first people in the real-time computing field to recognize and respond to this problem – I began in the early 1970's to perform both experimental research on, and transition of resulting technologies to, large complex distributed real-time computer systems for defense (see [About Me](#)) and then industrial automation. Only in the past several years have significant numbers of other real-time researchers begun to expand their horizons beyond static low-level real-time (for example, rate monotonic analysis) – thanks to the appearance of DARPA funding encouraging this expansion (e.g., the [Quorum](#) program).

Next is a brief self-contained [overview](#) of my formulation of the most fundamental concepts and terms in real-time computing. This formulation has been demonstrated to greatly facilitate the cost-effective construction of real-time computing systems that are far more complex (e.g., dynamic, adaptive, distributed) than can be constructed using traditional ill-defined real-time concepts.

Then, the material is covered in more detail – in particular, exploiting the expressiveness and generality of the *time/utility function* (ne'e time/value function) framework.

Next: [Real-Time Overview](#)



[Add to Favorites](#)



[Print Page](#)



[Download a PDF copy of this page](#)

[View Site Changes](#) | [XML](#) | [Site Updated 02/29/2004](#) | [Legal](#)