THE ART OF SYSTEMS ARCHITECTING

by

Eberhardt Rechtin
Principal, Center for Software Engineering
Emeritus Professor of Electrical Engineering Systems

10 March 1997
THE ART OF SYSTEMS ARCHITECTING

What is it?

What's a system?
A collection of elements which together produce a unique result.
An increasingly important subset: systems of systems

What is architecting?
The process of creating and building systems
Complexity: stakeholders, multi-disciplines, unboundedness

What is the art of systems architecting
The non-analytic complement to analytic systems engineering
The “practice” of systems synthesis and certification

Why systems architecting?
1. The realization that “something is missing” in systems integration
2. “Complexity is the major cause of overruns and performance deficiency”
   Multidisciplinary, multiple stakeholders, inconsistent standards and regulations
3. The inherent limitations of (analytic) engineering
   Deduction vs induction. The need for complete information and a defined scope.
Why now?

The scope of modern systems

Cold War "stability" and systems evolution

Post-Cold War transition to global competition and new architectures

The advent of real-time, software-intensive systems

The increasing rate of technological advance in many disciplines

Applications

Space and defense systems

Apollo, Shuttle, Space Station, Planetary exploration and the Deep Space Network

Communication satellites

Surveillance satellites: weather, intelligence, agriculture, resource management

Air defense, missile defense, naval battle groups, land-sea-air, logistics, C3I

Consumer goods and services

Personal computers, Internet, e-mail, public utilities, manufacturing, software,

social systems (transportation, education, health, welfare), entrepreneurial systems,

management systems
“TOOLS OF THE TRADE”

(Science and mathematics-based tools a “given”)
Algorithms, equations, tradeoffs, laws of physics and mathematics

Models
Form
Scale, block diagram
Functional
Threads and scenarios
Data and event flow
Performance (economics, sociological, political, user)

Heuristics
Descriptive
Murphy’s law. The Rubber Schedule. All the serious mistakes are made in the first day.
Prescriptive
Simplify! Use open architectures. More generally, keep your options open.
Task-oriented
Scoping and planning, modeling, prioritizing, aggregating, partitioning, integrating, certifying, assessing, and rearchitecting

Metaphors
Desk tops, optimization landscapes, search maps, hardware stores

Sources: