COMPUTER-AIDED SYSTEMS ENGINEERING
COMPUTER-AIDED SYSTEMS ENGINEERING

Dr. S. Andriole

SPC-94039-CMC

Version 01.00.00

JUNE 1994

This material is based in part upon work sponsored by the Advanced Research Projects Agency under Grant # MDA572-92-J-101K. The content does not necessarily reflect the position or the policy of the U.S. Government, and no official endorsement should be inferred.

This document accompanies a videotape of the same presentation recorded live at the Software Productivity Consortium in March 1994. It is recommended that the videotape be viewed with these view-graphs at hand.

Produced by the SOFTWARE PRODUCTIVITY CONSORTIUM under contract to the VIRGINIA CENTER OF EXCELLENCE FOR SOFTWARE REUSE AND TECHNOLOGY TRANSFER 59 Building 2214 Rock Hill Road Herndon, Virginia 22070
ABSTRACT

Computer-Aided Systems Engineering

Dr. Stephen J. Andriole
Drexel University

This presentation includes information concerning toolset requirements, current and projected tools, tool capabilities assessment and perceived critical deficiencies. Dr. Andriole presents a tools assessment matrix that demonstrates a mapping between toolset requirements and currently available systems engineering tools. He discusses the benefits and problems that organizations experience with tools. The timing and costs of investments in tools, how tools can be expected to perform and future expectations are presented.

Dr. Andriole explains why companies choose tools, identifies the most favorable environments and discusses common obstacles to successful CASE use. This video is intended for systems and software engineering development lead engineers, project managers and division managers working in the area of system and software engineering. Viewers will benefit by gaining information about systems engineering tools and automation, especially the value and capability of commercial off-the-shelf (COTS) tools. Dr. Andriole emphasizes the use of lower cost PC based (DOS or MAC) tools to support systems engineering activities including requirements modeling, simulation and prototyping, evaluation and trade-off analysis, testing and reliability, technology forecasting and risk analysis.

Dr. Andriole presents an overall framework for evaluating systems engineering tools based on how effectively the tools support the systems engineering process, followed by an overview of selected tools that are currently available.

Dr. Andriole is the Director of Information Systems Technology Laboratory at Drexel University. Prior to this he worked for several years at the Defense Advanced Research Projects Agency and was a professor at George Mason University. Dr. Andriole has broad expertise in the area of systems analysis and systems engineering of software intensive systems. He has published several articles related to systems engineering tools.
Computer-Aided Systems Engineering

An Assessment of Current Practices, Tools & Trends

Stephen J. Andriole
The Assessment Framework

- Systems Engineering
- The State-of-the-Practice of Computer-Aided Systems Engineering
- The Generic Systems Engineering Framework
- Phase-by-Phase Organization & Categorization
- Criteria-Based Evaluation & Assessment --- The Matrix
- COTS/CASE Tools Sampler
- SPC COTS/CASE "Products" & Services ...
The State-of-the-Practice of Computer-Aided Systems Engineering

- "Practiced" for Years by Systems Engineers

- Opportunistic -- Not Characterized by the Same Zeal or Investments that Characterize the Computer-Aided Software Engineering Movement

- In Many Respects -- Further Integrated into the Other-Than-Software-Intensive Design & Development Process Than in the Software-Intensive Systems Design & Development
The State-of-the-Practice of Computer-Aided Systems Engineering (Continued)

- Practiced as Pieces of a Greater Whole:
  - Cost-Benefit Modeling
  - Technology Forecasting
  - Project Management
  - Resource Allocation
  - Cost Estimation
  - Trade-Off Analysis ...
  - No "I-CASE" ... Yet ...

- "Attached" to Life Cycle Phases
The "Generic" Systems Engineering Framework

- May Be Described -- Like the Software (Systems) Engineering Life Cycle -- as a "Conversion Process" -- From Fuzzy & Ill-Understood Requirements to Detailed Design & Measurable Performance

- Well-Supported by "Standards", Such as Blanchard's Framework & 499A/B

- Characterized by:
  - Objectives (by Phase & Overall)
  - Methods
  - Tools
Systems Engineering

- The Process by Which we Convert Complex Requirements into Working, Maintainable Systems is Part of the Discipline and Structure Known as Systems Engineering

- While there are All Sorts of Variations on this Theme, Systems Engineering Refers to Those Processes and Activities that Together Define a Life Cycle that Follows a System Through Crude Concept Development, Testing and Evaluation, and Eventual Retirement
Systems Engineering


- More Recently, Sage (1992), Blanchard (1991), Eisner (1988), and Chapman, Bayhill and Wymore (1992) have Refined the Methods, Tools Techniques, Activities, Functions and Purpose of Systems Engineering

- Along the Way an Overarching Department of Defense Standard Evolved
Systems Engineering

- The Essence of the Systems Engineering Process is Requirements --- Design --- Development Efficiency

Systems Engineering

- Eisner defines systems engineering as "an iterative process of top-down synthesis, development, and operation of a real-world system that satisfies, in a near optimal manner, the full range of requirements for the system.” Eisner describes the systems engineering process as consisting of 25 “elements”
Systems Engineering

1. Requirements Analysis
2. Requirements Allocation
3. Functional Analysis
4. Functional Allocation
5. Specification Analysis
6. Specification Development
7. Preliminary Design
8. Interface Definition
9. Schedule Development
10. Preliminary Cost-Analysis
11. Technical Performance Measurement
12. Trade-Off/Alternative Analysis
13. Pre-Planned Product Improvement
14. Final Design
15. Schedule Update
16. Cost Update
Systems Engineering

17. Fabrication
18. Coding
19. Preliminary Testing
20. Debugging & Reconfiguration
21. Testing & Integration
22. Updates

   A. Schedule
   B. Cost
   C. Technical Performance Measurement

23. Documentation
24. Training
25. Production
Systems Engineering

- 499A

"Systems engineering is the ... logical sequence of activities and decisions transforming an operational need into a description of system performance parameters and a preferred system configuration ... systems engineering is the application of scientific and engineering efforts to (a) transform operational need into a description of system performance parameters and a system configuration through the use of an iterating process of definition, synthesis, analysis, design, test, and evaluation; (b) integrate related technical parameters and ensure compatibility of all physical, functional, and program interfaces in a manner that optimizes the total system definition and design; (c) integrate reliability, maintainability, safety, survivability, human, and other such factors into the total engineering effort to meet cost, schedule, and technical performance objectives."
Systems Engineering

- The Revised (Draft) Standard (DOD, 1992) -- 499B --
  Describes the Systems Engineering Process as Follows:

  "A comprehensive, iterative problem solving process that is used to: (a) transform validated customer needs and requirements into a life-cycle balanced solution set of system product and process designs, (b) generate information for decision-makers, and (c) provide information for the next acquisition phase"
Systems Engineering Goals
(After Sage)

- All (Life Cycle) Encompassing
- Problem Understanding
- Communication
- Early Capture of Design & Implementation Needs
- Bottom-Up & Top-Down Design & Development
- Alternative Systems Management Approaches
- Process & Product Quality Assurance
- Product Evolution
- Support for Configuration Management Standards
- Support for Automated Design & Development Aids
- Teachable & Transferable Methodology
- All phase Definition & Documentation
- Product Functionality, Revisability & Transitioning
- Support System Product Development & System User Organizations
Systems Engineering Management Plan (SEMP)

- At the Core of the Planning and Management Process lies the SEMP that Requires the Production of Documentation that can be Reviewed and Assessed. This Documentation Includes at Least the Following:

  - Project Management Schedules
  - Product Cost Estimation
  - Timeline Analysis Sheets
  - Requirements Allocation Sheets
  - Work Breakdown Structures
  - Technical Performance Measures
  - Human Factors Engineering Plan
  - Risk Management Plan
  - Data/knowledge Management Plan
  - Maintainability Plan
  - User Manuals
- Software Requirements Specifications (Preliminary & Detailed)
- Software Design Specifications
- Test & Evaluation Master Plan
- Configuration Management Plan
- Production/Manufacturing Plan
- Total Quality Management Plan
The Waterfall Model Of The Software Life Cycle

System Feasibility
Validation
Software Plans and Requirements
Validation
Product Design
Verification
Detailed Design
Verification
Code
Unit Test
Integration
Product Verification
Implementation
System Test
Operations and Maintenance
Revalidation

(Royce)
Generic Systems Engineering Process

1. State Needs for System
2. Develop Goals and Objectives
3. Define System Requirements
4. Write System Specifications

5. Synthesize System Alternatives (Life Cycle)
6. Assess Possible Constraints

7. Analyze System Alternatives
8. Formulate Evaluation Criteria

9. Update Specifications
10. Build, Test, and Accept System
11. Document and Install System
12. Operate, Maintain, and Monitor System
13. Modify and Upgrade System

Requirements & Specifications
Design and Analysis
Construction and Operation

(Eisner)
Generic Systems Engineering Process

Input Requirements
- Objectives
- Environment
- Constraints
- Measures of Effectiveness

Technology Selection Factors
- Hardware
- Software
- Reliability
- Maintainability
- Human Factors
- Security
- Safety
- Standardization
- Productability
- Transportability
- Computer Resources

FUNCTIONAL ANALYSIS

SYNTHESIS

WILL ALTERNATIVES WORK?

EVALUATION AND DECISION (TRADE-OFF)

ACCEPTABLE SOLUTION

Description of System Elements
- Facilities
- Computer Hardware
- Computer Software
- User-Computer Interface
- Analytical Methods

(Seng)
Development of Functional Flow Block Diagrams

TOP-LEVEL DIAGRAM

FIRST LEVEL: FLIGHT MISSION

SECOND-LEVEL DIAGRAM

SECOND LEVEL: 4.0 PERFORM MISSION OPERATIONS

THIRD-LEVEL DIAGRAM

THIRD LEVEL: 4.8 ACQUIRE PAYLOAD DATA
Trade-Off Analysis Methodology

1. Define Objectives and Requirements
2. Identify Alternatives
3. Formulate Selection Criteria
4. Weight Criteria
5. Prepare Utility Functions
6. Evaluate Alternatives
7. Perform Sensitivity Check
8. Select Preferred Alternatives
9. Execute Decision
10. Eliminate Sensitivities
A Simplified Structure Of A Radio Evaluation Model

Technical System Utility

Technical Performance

Dependability

Channels + Interface + Technical Flexibility + Interchangeability

Reliability + Availability

Dimensions of System Availability and Performance

System Parameters

Retransmission + R/T + EMC

Utility Functions

[Graphical representation of the structure with nodes and arrows indicating relationships between parameters and dimensions]
A Systems Engineering Management Structure

SYSTEMS ENGINEERING MANAGEMENT

Technical Program Planning and Control
- Work Breakdown Structure and Specification Tree
- Program Risk Analysis
- System Test Planning
- Decision and Control Process
- Technical Performance Measurement
- Technical Reviews
- Vendor Reviews
- Work Authorization
- Documentation Control

Systems Engineering Process
- Mission Requirements Analysis
- Functional Analysis
- Allocation
- Synthesis
- Logistic Engineering
- Life-Cycle Cost Analysis
- Optimization
- Production-Engineering Analysis
- Generation of Specifications

Engineering Specialty Areas
- Reliability, Maintainability, and Availability Analysis
- Human Factors
- Safety
- Security
- Configuration Control
- Interface Definition and Control
- Interoperability
- Training
- Environmental Compatibility
- Other Specialty Areas

(Elsner)
Time Scope

Level of Involvement

Pre-Development  Initial Development  Operation & Modification  D:missioning

SYSTEMS ENGINEERING

SOFTWARE ENGINEERING
# Systems Engineering "Versus" Software Engineering

<table>
<thead>
<tr>
<th></th>
<th>Systems Engineering</th>
<th>Software Engineering</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alternative Design Methods</td>
<td>H</td>
<td>M</td>
</tr>
<tr>
<td>Multidisciplinary Orientation</td>
<td>H</td>
<td>L</td>
</tr>
<tr>
<td>Requirements Analysis &amp; Prototyping</td>
<td>H</td>
<td>M</td>
</tr>
<tr>
<td>Criteria-Based Trade-Off Analysis</td>
<td>H</td>
<td>L</td>
</tr>
<tr>
<td>Detailed Software Specifications</td>
<td>M</td>
<td>H</td>
</tr>
<tr>
<td>Optimal Software Production</td>
<td>M</td>
<td>H</td>
</tr>
<tr>
<td>Methods Integration (Re-) Planning</td>
<td>M</td>
<td>L</td>
</tr>
<tr>
<td>Education &amp; Training Curricula</td>
<td>M</td>
<td>M</td>
</tr>
<tr>
<td>Artifact Profiling</td>
<td>M</td>
<td>M</td>
</tr>
<tr>
<td>Applications Range Assessment</td>
<td>M</td>
<td>M</td>
</tr>
<tr>
<td>Measures &amp; Standards</td>
<td>M</td>
<td>M</td>
</tr>
<tr>
<td>Processes/Metrics Introspection</td>
<td>M</td>
<td>M</td>
</tr>
</tbody>
</table>
Computer-Aided Systems Engineering

"Customized" Modeling & Analysis Tools

Generic Modeling & Analysis Tools

"Conventional" CASE Tools
Criteria-Based Evaluation & Assessment --> The Tools Matrix

- Tools Can Be Assessed According to Their:
  - "Fit" (with a Systems Engineering Phase, Method, or Objective
  - Cost (to Include Acquisition, Training & Use)
  - Quality of Output
  - Ease of Use
  - Robustness
  - Support ...
<table>
<thead>
<tr>
<th>3-D Computer Package</th>
<th>Platform</th>
<th>Operating System</th>
<th>Life Cycle Phase</th>
<th>Experience</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>3-D Graphics Toolkit</td>
<td>IBM/Clone</td>
<td>DOS</td>
<td>Prototyping/Modeling</td>
<td>Novice</td>
<td>89</td>
</tr>
<tr>
<td>4th Dimension</td>
<td>Macintosh</td>
<td>Macintosh</td>
<td>Prototyping/Modeling</td>
<td>Intermediate</td>
<td>80</td>
</tr>
<tr>
<td>ABC Flowchart</td>
<td>IBM/Clone</td>
<td>Windows</td>
<td>System Transfer</td>
<td>Intermediate</td>
<td>895</td>
</tr>
<tr>
<td>ACD-Action Chart Diagrammer</td>
<td>IBM/Clone</td>
<td>DOS</td>
<td>Prototyping/Modeling</td>
<td>Intermediate</td>
<td>295</td>
</tr>
<tr>
<td>ACPVision</td>
<td>Macintosh</td>
<td>Macintosh</td>
<td>Requirements Analysis</td>
<td>Intermediate</td>
<td>495</td>
</tr>
<tr>
<td>Acta Advantage</td>
<td>Macintosh</td>
<td>Macintosh</td>
<td>Requirements Analysis</td>
<td>Intermediate</td>
<td>495</td>
</tr>
<tr>
<td>Adobe Illustrator 88</td>
<td>Macintosh</td>
<td>Macintosh</td>
<td>Requirements Analysis</td>
<td>Novice</td>
<td>129</td>
</tr>
<tr>
<td>AEC Information Manager</td>
<td>Macintosh</td>
<td>Macintosh</td>
<td>Prototyping/Modeling</td>
<td>Novice</td>
<td>50</td>
</tr>
<tr>
<td>Aege Showcase FIX</td>
<td>Macintosh</td>
<td>Macintosh</td>
<td>Other</td>
<td>Intermediate</td>
<td>495</td>
</tr>
<tr>
<td>Agenda</td>
<td>IBM/Clone</td>
<td>DOS</td>
<td>Project Management</td>
<td>Casual</td>
<td>695</td>
</tr>
<tr>
<td>Aktus Persuasion</td>
<td>Macintosh</td>
<td>Macintosh</td>
<td>Project Management</td>
<td>Intermediate</td>
<td>395</td>
</tr>
<tr>
<td>Anatool</td>
<td>Macintosh</td>
<td>Macintosh</td>
<td>Project Management</td>
<td>Intermediate</td>
<td>395</td>
</tr>
<tr>
<td>Animation Works</td>
<td>Macintosh</td>
<td>Macintosh</td>
<td>Project Management</td>
<td>Intermediate</td>
<td>495</td>
</tr>
<tr>
<td>AppMaker, The Application Gen</td>
<td>Macintosh</td>
<td>Macintosh</td>
<td>Prototyping/Modeling</td>
<td>Intermediate</td>
<td>925</td>
</tr>
<tr>
<td>ATLAS MapMaker</td>
<td>Macintosh</td>
<td>Macintosh</td>
<td>Requirements Analysis</td>
<td>Intermediate</td>
<td>200</td>
</tr>
<tr>
<td>Authorware Professional</td>
<td>Macintosh</td>
<td>Macintosh</td>
<td>Prototyping/Modeling</td>
<td>Casual</td>
<td>295</td>
</tr>
<tr>
<td>Black &amp; White Movies</td>
<td>Macintosh</td>
<td>Macintosh</td>
<td>Requirements Analysis</td>
<td>Casual</td>
<td>495</td>
</tr>
<tr>
<td>Blues</td>
<td>Macintosh</td>
<td>Macintosh</td>
<td>Prototyping/Modeling</td>
<td>Expert</td>
<td>8000</td>
</tr>
<tr>
<td>CA-ESTIMACS</td>
<td>IBM/Clone</td>
<td>DOS</td>
<td>Requirements Analysis</td>
<td>Novice</td>
<td>50</td>
</tr>
<tr>
<td>Canvas</td>
<td>IBM/Clone</td>
<td>DOS</td>
<td>Other</td>
<td>Novice</td>
<td>1875</td>
</tr>
<tr>
<td>CASE:PM</td>
<td>IBM/Clone</td>
<td>OS2</td>
<td>Software Selection/Design</td>
<td>Novice</td>
<td>75</td>
</tr>
<tr>
<td>CASE:W</td>
<td>IBM/Clone</td>
<td>DOS</td>
<td>Prototyping/Modeling</td>
<td>Novice</td>
<td>195</td>
</tr>
<tr>
<td>Chart Master</td>
<td>IBM/Clone</td>
<td>DOS</td>
<td>Other</td>
<td>Casual</td>
<td>80</td>
</tr>
<tr>
<td>Charts Unlimited</td>
<td>IBM/Clone</td>
<td>DOS</td>
<td>Requirements Analysis</td>
<td>Casual</td>
<td>495</td>
</tr>
<tr>
<td>Choreographer</td>
<td>IBM/Clone</td>
<td>OS2</td>
<td>Prototyping/Modeling</td>
<td>Casual</td>
<td>195</td>
</tr>
<tr>
<td>Clarion Personal Developer</td>
<td>IBM/Clone</td>
<td>DOS</td>
<td>Prototyping/Modeling</td>
<td>Casual</td>
<td>199</td>
</tr>
<tr>
<td>CoCoPro</td>
<td>Macintosh</td>
<td>Macintosh</td>
<td>Requirements Analysis</td>
<td>Experienced</td>
<td>995</td>
</tr>
<tr>
<td>College Display Utilities</td>
<td>IBM/Clone</td>
<td>DOS</td>
<td>Prototyping/Modeling</td>
<td>Intermediate</td>
<td>995</td>
</tr>
<tr>
<td>ColorX VGA Paint</td>
<td>IBM/Clone</td>
<td>DOS</td>
<td>Prototyping/Modeling</td>
<td>Intermediate</td>
<td>35</td>
</tr>
<tr>
<td>Company Ladder</td>
<td>IBM/Clone</td>
<td>DOS</td>
<td>Project Management</td>
<td>Experience</td>
<td>7500</td>
</tr>
<tr>
<td>Cost Benefit Models</td>
<td>IBM/Clone</td>
<td>DOS</td>
<td>Intermediate</td>
<td>495</td>
<td></td>
</tr>
<tr>
<td>Cost/Schedule Workshop (CSW)</td>
<td>IBM/Clone</td>
<td>DOS</td>
<td>Intermediate</td>
<td>169</td>
<td></td>
</tr>
<tr>
<td>Course Builder</td>
<td>Macintosh</td>
<td>Macintosh</td>
<td>Project Management</td>
<td>Intermediate</td>
<td>495</td>
</tr>
<tr>
<td>DataPro</td>
<td>IBM/Clone</td>
<td>DOS</td>
<td>Requirements Analysis</td>
<td>Casual</td>
<td>80</td>
</tr>
<tr>
<td>DataPro</td>
<td>IBM/Clone</td>
<td>DOS</td>
<td>Requirements Analysis</td>
<td>Casual</td>
<td>199</td>
</tr>
<tr>
<td>Decision Analysis</td>
<td>IBM/Clone</td>
<td>DOS</td>
<td>Requirements Analysis</td>
<td>Casual</td>
<td>40</td>
</tr>
<tr>
<td>Decision Analysis By TrueAge</td>
<td>Macintosh</td>
<td>Macintosh</td>
<td>Requirements Analysis</td>
<td>Casual</td>
<td>595</td>
</tr>
<tr>
<td>Decision Analysis Techniques</td>
<td>Macintosh</td>
<td>Macintosh</td>
<td>Requirements Analysis</td>
<td>Casual</td>
<td>249</td>
</tr>
<tr>
<td>Decision Analysis By TrueAge</td>
<td>Macintosh</td>
<td>Macintosh</td>
<td>Requirements Analysis</td>
<td>Casual</td>
<td>195</td>
</tr>
</tbody>
</table>

Note: The table lists various software packages and their platforms, operating systems, and associated life cycle phases and experience levels. The column labeled 'P' appears to represent a numerical value, possibly related to a weight or priority score.
COTS/CASE Tools Sampler

- Tools Not Usually Associated with Software Engineering + Some of Those That Are!!!

- Tools That Support Methods & Objectives of the Systems Engineering Life Cycle Phases

- Tools That Support:
  - Modeling, Charting & Diagramming
  - Analysis (eg, Cost-Benefit, Estimation)
  - Documentation & "Auditing"
  - Management
  - Technology Assessment
  - Testing & Evaluation ...
A Taxonomy Of Analytical Methods & Sub-Methods

1. SUBJECTIVE ASSESSMENT
   - Probability Assessment
   - Anomalous Event Matrix
   - Brainstorming
   - Synectics
   - Delphi
   - Historical Analogy
   - Comparative Analysis

2. STRUCTURED QUALITATIVE
   - Cost-Benefit Analysis
   - Change Signals Monitoring
   - Leading Indicators
   - Cross-Impact Analysis
   - Bayesian Updating
   - Probability Trees
   - Morphological Analysis

3. TIME SERIES/EXTRAPOLATION
   - Growth Curves, Trends, & Cycles
   - Smoothing Methods
   - Box-Jenkins
   - Robust Extrapolation Methods

4. STOCHASTIC/PROBABILISTIC
   - Descriptive Profiling
   - Correlation
   - Simple Regression
   - Multiple Regression
   - Game Theory
   - Inventory Theory

5. STATISTICAL/OPERATIONS RESEARCH
   - Leading Indicators
   - Econometric Models
   - System Dynamics Models (Simulation)

6. CAUSAL MODELS

7. INFORMATION MANAGEMENT
   - Data Base Organization:
     - DBM/OSMS
   - Man-Machine Communication:
   - Software For Analysis

8. ARTIFICIAL INTELLIGENCE
   - Expert Systems
   - Natural Language Processing
   - Others

QUALITATIVE

HYBRID/BOOTSTRAPPING

QUANTITATIVE

INFORMATION SCIENCE
Life Cycle Phase-by-Phase Organization & Categorization

- Pre-Phases
  - Project Management Tools
  - Simple Graphics Tools
  - Modeling & Analysis Tools
  - "Visualization" As A Way of Life

- Early Phases & Activities
  - Requirements, Requirements, Requirements
  - Requirements Models, Tools & "Environments"
  - Trade-Off Analyses
  - Constraint Analysis ...
Life Cycle Phase-by-Phase Organization & Categorization (Continued)

• Mid-Phases & Activities
  - Specification (from Multiple Perspectives)
  - On-Going Requirements Traceability
  - Design (from Multiple Perspectives)
  - Development (Often Evolutionary)

• Later Phases & Activity
  - T&E; V&V
  - On-Going Requirements Traceability
  - Documentation (Throughout)
  - Configuration Management
Exemplar CAS(ystems)E Tools

- Requirements Modeling
  - QFD Capture
  - RDD
  - DOOR
  - TopDown
  - DATA
  - IDEF0
  - Inspiration
  - Systems Engineering Design Software (SEDSO)
  - Conventional CASE Tools
  - Many CAD/CAM/CALS Tools
  - Some Evaluation & Trade-Off Analysis Tools
Exemplar CAS(ystem)E Tools

- Simulation & Prototyping
  - Extend
  - iThink
  - MicroSaint
  - MetaDesign
  - Skylights
  - ToolBook
  - Show Partner
  - TAE+
  - Access
  - LabView
  - Design/IDEF/IDEF0
  - Visual Basic
  - Prograph
Exemplar CAS(systems)E Tools

- Evaluation & Trade-Off Analysis
  - Logical Decision
  - Expert Choice
  - DecisionMap
  - Decision Analyst
  - Lightyear
  - ADAM 2
  - Arborist
  - COMPARE!
  - Best Choice 3
  - Criterium
  - Crystal Ball
  - David
  - Decision Pad
  - Equity
Exemplar CAS(systems)E Tools

- Expression Tree
- HIVIEW
- SuperTree
Exemplar CAS(systems)E Tools

- Testing & Reliability
  - RPP
  - PC Availability
  - PC Predictor
  - Tiger Computer Program(s)
  - Mechanical Reliability Prediction Program (MPP)
  - Maintainability Effectiveness Analysis Program (MEAP)
  - Optimum Repair Level Analysis (ORLA) Model
  - Equipment Designer's Cost Analysis System (EDCAS)
  - Network Repair Level Analysis (NRLA)
  - OPUS Model
  - VMETRIC
Exemplar CAS(systems)E Tools

- Systems & Logistics Integration Capability (SLIC)
- Life Cycle Cost Calculator (LCCC)
- Cost Analysis Strategy Assessment (CASE)
- Life-Cycle Model for Defense Material Systems
Exemplar CAS(systems)E Tools

- Technology Forecasting
  - 4CAST/2
  - Autocast
  - Forecast!
  - GLIM
  - MTS
  - NCSS
  - SAS
  - SiBYL/Runner
  - STORM
Exemplar CAS(systems)E Tools

- Risk Analysis
  - @Risk
  - RiskWatch
  - Automated Risk Evaluation System (ARES)
  - Bayesian Decision Support System (BDSS)
  - The Buddy System
  - Continuous Risk
  - Los Alamos Vulnerability & Risk Assessment (LAVA)
Exemplar CAS(systems)E Tools

- Prototyping Tools for Software Systems Engineering
- Tool "Requirements"
- Exemplar Tool Types
- Specific Tools ...
Impact

- Software Support For All Phases of Life Cycle
- Capability to Store & Model Requirements
  Data --> Prototypes --> Specifications
  Designs
- "Audit Trail" of Design/Development Process
- Capability to Store Designs, Models, Prototypes & Modules & Reuse the Concepts, Data, Models & Software
Interactive Requirements Modeling & Prioritization & System Concept Design

for

Rouge River Decision Support
Interactive Requirements Modeling for Rouge River Decision Support
The Process

- Collect Requirements Data Via Individual & Group Discussions & Via Codified Requirements Data

- Document Requirements Data -- Not in Elaborate "Specs" -- But in a Form that Lends Itself to Analysis & Modification

- Organize the Requirements Data in Alternative Forms for Iteration -- Such as in Simple Outline Form & in a Hierarchical Form

- Assess, Rank-Order, Trade-Off -- "Reconcile" Requirements

- Identify Off-the-Shelf & Existing Special Purpose Applications Programs & Implementation Environments

- Match the Requirements to the Existing Systems

- Develop a System Requirements Specification that Lies at the Intersection of Requirements, Existing Systems & Constraints
Some Special Assumptions & Features of the Process

• The Process Assumes the Value of "Participatory Design"

• The Process Assumes the Value of Iterative Design & Prototyping -- to a Point!

• The Process Assumes the Value of a Life Cycle-Driven System Development Methodology

• The Process Assumes the Need to Make Requirements Explicit & to "Model" Requirements to Determine What are the "Most," "Moderately," and "Not Quite as" Important Requirements

• The Process Assumes the Stupidity of Re-Inventing the Wheel

• Requirements will Be Captured, Modeled & Communicated to All "Stakeholders" During the Process
Requirements Modeling --> Prototyping
Process --> Management Plan

Step 1: Gather/Elicit Requirements Data
Step 2: Model Requirements Via Working Groups
Step 3: Assess User/Technological/"Political" Constraints

Step 4: Assess Existing Models, Tools & Environments
Step 5: Develop Alternative (DSS) Designs
Step 6: Develop Criteria to Evaluate Designs

Step 7: Identify "Best" Candidate Design
Step 8: Develop Detailed Management Plan

Step 9: Proceed with Prototyping Process
System Concept Evaluation Process

1. Develop Overall Objectives
2. Define Functional & Design Requirements
3. Model Requirements
4. Identify Alternative System Concepts
5. Define & Weight Criteria
6. Score the Concepts Vis-a-Vis the Criteria
7. "What-If" Sensitivity Analyses
8. Rank-Ordered Alternative System Concepts
9. Prototyping
**Decision Outline on DSS H/S Architecture**

Alternatives:
- System Concept #1
- System Concept #2
- System Concept #3

Factors:
- DSS H/S Architect: e

1. Functional Requirements

2. Analytical Requirements

3. Flow Analysis
   4. Whole River Simulate
   4. Threshold Analysis
   4. Quasi-RT Analysis
   4. Realtime Analysis

3. Map Partitioning
   4. Scenario-Driven
   4. Clutter/De-clutter

3. Querying
   4. Cost-Result Analyses
   4. Comparative Analyses
   4. Sensitivity Analyses

2. Communicative Requirements

3. Advocacy Requirements
   4. Regional Group Support
   4. Local Support
3 Policymkg Requirm'ts

4 Remote PMKG Support

4 Local Support

5 Investment Strategy

5 In-Stream Analysis

3 Educatn'l Requirm'ts

4 Water Pollut'n Educat

4 General Pollution

4 RR Basin Pollution

1 Design Requirements

2 H/S Architecture

3 Optimize Hardware

3 Optimize Models

3 Optimize OS Trends

2 Local Modeling

3 Local/Remote Network

2 Design to Cost

2 Design to Schedule
<table>
<thead>
<tr>
<th>Alternative</th>
<th>Score</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Concept #3</td>
<td>98.55</td>
<td>Concept for PC/NT-based DSS with full networking</td>
</tr>
<tr>
<td>System Concept #2</td>
<td>75.84</td>
<td>Concept for UNIX-based system w modest networking</td>
</tr>
<tr>
<td>System Concept #1</td>
<td>57.84</td>
<td>Concept that calls for UNIX workstation-based DSS</td>
</tr>
</tbody>
</table>

Comments: "Optimal" hardware/software architecture
SYSTEM CONCEPT 1

- SUN/UNIX
- ARC/INFO
- SPATIAL/TEMPORAL INFORMATION
- INTERACTIVE VISUALIZATION
Decision Results on DSS H/S Architecture

<table>
<thead>
<tr>
<th>System Concept</th>
<th>Best</th>
<th>Worst</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1</td>
<td>57.84</td>
<td></td>
</tr>
<tr>
<td>#2</td>
<td>75.84</td>
<td></td>
</tr>
<tr>
<td>#3</td>
<td>98.55</td>
<td></td>
</tr>
</tbody>
</table>

COMMENTS: "Optimal" hardware/software architecture
### Decision Comparison on Communication's Requirements

<table>
<thead>
<tr>
<th>Advocacy Requirements</th>
<th>Education Requirements</th>
<th>Policy-making Requirements</th>
</tr>
</thead>
</table>

**Comments:** Requirements that describe communication functions

- System Concept #1
- System Concept #2
- System Concept #3
## Decision Comparison on Design Requirements

<table>
<thead>
<tr>
<th>H/S Architecture</th>
<th>Design to Schedule</th>
<th>Design to Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Best</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Worst</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**COMMENTS:** Requirements that refer to design considerations

- System Concept #1
- System Concept #2
- System Concept #3
Decision Comparison on DSS H/S Architecture

<table>
<thead>
<tr>
<th></th>
<th>System Concept #1</th>
<th>System Concept #2</th>
<th>System Concept #3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Best</td>
<td>57.84</td>
<td>75.84</td>
<td>98.55</td>
</tr>
<tr>
<td>Worst</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

COMMENTS: "Optimal" hardware/software architecture

- [ ] Funcrl Requirements
- [ ] Design Requirements
Trends

• Toward I-CASE
• Toward Greater Distribution
• Toward Automation ...
Key References

• Howard Eisner, Computer-Aided Systems Engineering, Prentice-Hall, 1988


• Andrew P. Sage, Systems Engineering, Wiley-Interscience, 1992


• Odean Bowler, et al., Requirements Analysis & Design Tools Report, Software Technology Support Center, 1992

• Chris Sittenauer, et al., Re-Engineering Tools Report, Software Technology Support Center, 1992
Questions or comments on content should be directed to:

Dr. Stephen J. Andriole
715 Cornerstone Lane
Bryn Mawr, PA 19010
andriole@DUVM.OCS.Drexel.Ed
(215) 525–5874

Or to:

Mary Skipp
Software Productivity Consortium
2214 Rock Hill Road
Herndon, VA 22070
skipp@software.org
(703) 742–7298

Send feedback on the Consortium's Video Program and orders for video products to:

Technology Transfer Clearinghouse
Software Productivity Consortium
2214 Rock Hill Road
Herndon, VA 22070
brewer@software.org
(800) 827–4772