

# Evaluation Criteria For Satellite Ground Systems

by

Charles B. Simmons

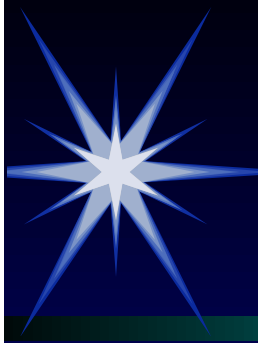
Information Technology Department

Computer Systems Division

The Aerospace Corporation

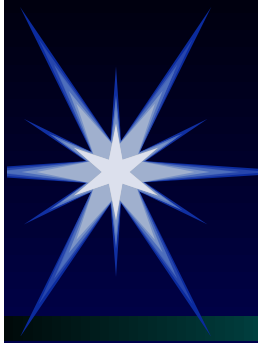
El Segundo, CA 90245

[simmons@aero.org](mailto:simmons@aero.org)



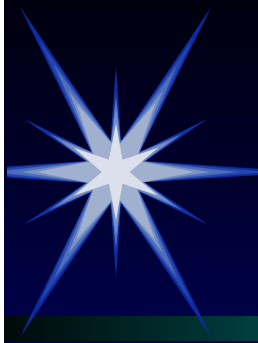
# Outline

- Background
- Definition of Architecture Evaluation
- Architecture Evaluation Methods
- Sample of Evaluation Criteria for Satellite Ground Systems (SGS)
- Example SGS Evaluation
- Summary & Status



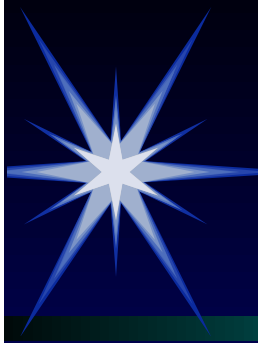
# Background

- Aerospace collaborating with USC on EDCS program
  - Exploring the use of EDCS technologies within the SGS domain
- Aerospace tasks
  - Investigated Architecture Definition Languages (ADLs) to support evaluation of SGS architectures
  - Develop evaluation criteria for SGS architectures



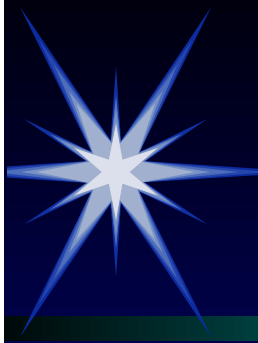
# Evaluation Criteria Task

- Developing a set of common domain-specific evaluation criteria (EC) for SGS
- Exploring evaluation methodologies
- EC based on knowledge of domain experts
- Following guidelines outlined in SEI Technical Report *Recommended Best Industrial Practice for Software Architecture Evaluation*, Jan. 1997
- Anticipated Result:
  - Evaluation Criteria & Methodology Report



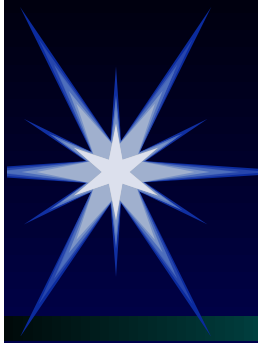
# Evaluation Criteria Defined

- Architecture Evaluation Criteria are:
  - *A set of predetermined questions and methods structured to determine important attributes of a given system*
- Architecture Evaluation is:
  - *The process of uncovering important attributes about the architecture of a given system*



# Why Evaluate Architecture?

- Detect problems early
- Compare & contrast candidate systems
- Increase understanding of system architecture
- Capture design rationale
- Clarification and prioritization of requirements
- Increases organizational learning



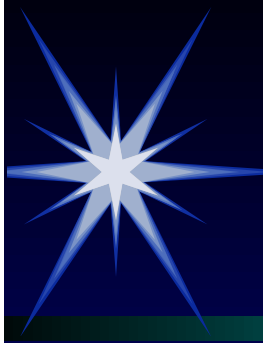
# Types of Architecture Evaluation

- Categories and types of architecture evaluation techniques from the SEI report:
  - Evaluation Method
    - Questionnaire
    - Checklist
    - Scenarios
    - Metrics
    - Prototype, Simulation, Experiment
  - Generality
    - General
    - Domain Specific
    - System Specific
  - Level of Detail
    - Coarse
    - Fine
  - Phase
    - Early
    - Middle
  - What is Evaluated
    - Artifact
    - Process



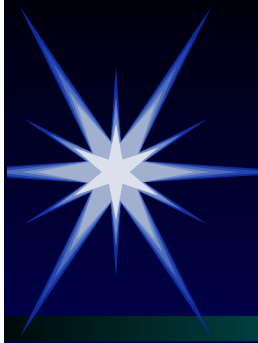
# Comparison of Categories of Evaluation Techniques

<b>Review Method</b>	<b>Generality</b>	<b>Level of Detail</b>	<b>Phase</b>	<b>What is Evaluated</b>
Questionnaire	general	coarse	early	artifact process
Checklist	domain-specific	varies	middle	artifact process
Scenarios	system-specific	medium	early - middle	artifact
Metrics	general or domain-specific	fine	middle	artifact
Prototype, Simulation, Experiment	domain-specific	varies	early	artifact



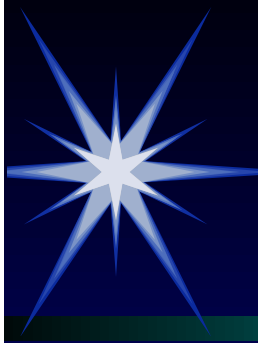
# Methods and Examples

- **Questionnaire** - *General questions that apply to all architectures. Used in early phase of life cycle.*
- **Examples:**
  - Is there a project architect?
  - Does the development team understand the architecture?
  - Is there a high level architecture description diagram?
  - Are components defined for the architecture?
  - Does the architecture depend on alternatives of the COTS product or OS?
  - Does software component definition come from the hardware division?
  - Is the architecture adhered to when you change the system?



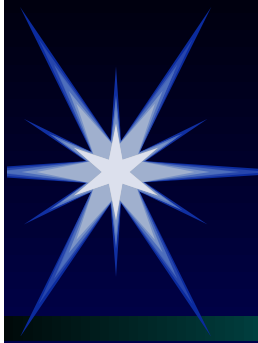
# Methods and Examples

- **Checklist** - *questions focusing on particular qualities of the system. These tend to be more domain specific.*
- **Examples for SGS domain:**
  - Does the architecture describe the architectural mechanism that allow components of the system to communicate?
  - Does the architecture specify standard interfaces between components?
  - Is the HMI encapsulated?
  - Is there consistency between static diagrams and the diagrams expressing dynamic behaviors?
  - Will a single component failure result in loss of telemetry or commanding?
  - Does the architecture support time-critical commanding?
  - Does the architecture allow components to receive telemetry synchronously?
  - What is degree of independence of application functions within components. E.g. telemetry - commanding?



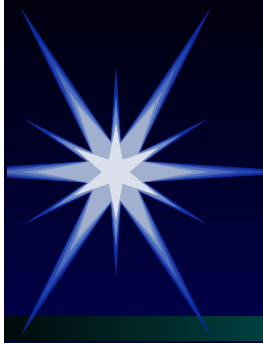
# Rationale Capture

- **Question:** Does the architecture support guaranteed message delivery?
- **Rationale:** In the event of a network or system failure, certain classes of messages (such as time critical commands) must have guaranteed delivery.
- **Warning Signs:** Many middleware products do not provide this capability. Adding this capability to an existing infrastructure will be expensive.
- **Weight (1-10):** 6



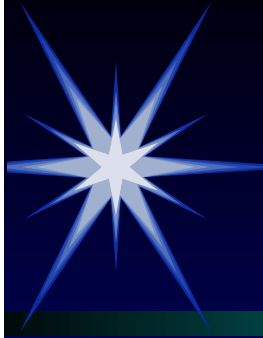
# Methods and Examples

- **Scenarios** - *a sequence of steps involving the use or modification of the system. Characterizes how well a particular architecture responds to demands placed on it within a context.*
- Software Architecture Analysis Method (SAAM)
- Good for evaluating quality attributes:
  - Extensibility, Performance, Scalability, Testability, Security
- Examples for SGS domain:
  - Time Critical Commanding scenario:
    - Monitor telemetry in realtime
    - Detect command constraint
    - Send command within time window
    - Monitor vehicle response to command
    - Notify operator of command status
  - Component replacement scenario:
    - COTS product for the anomaly management function must be added to the architecture to support constraint triggered commanding.



# Methods and Examples

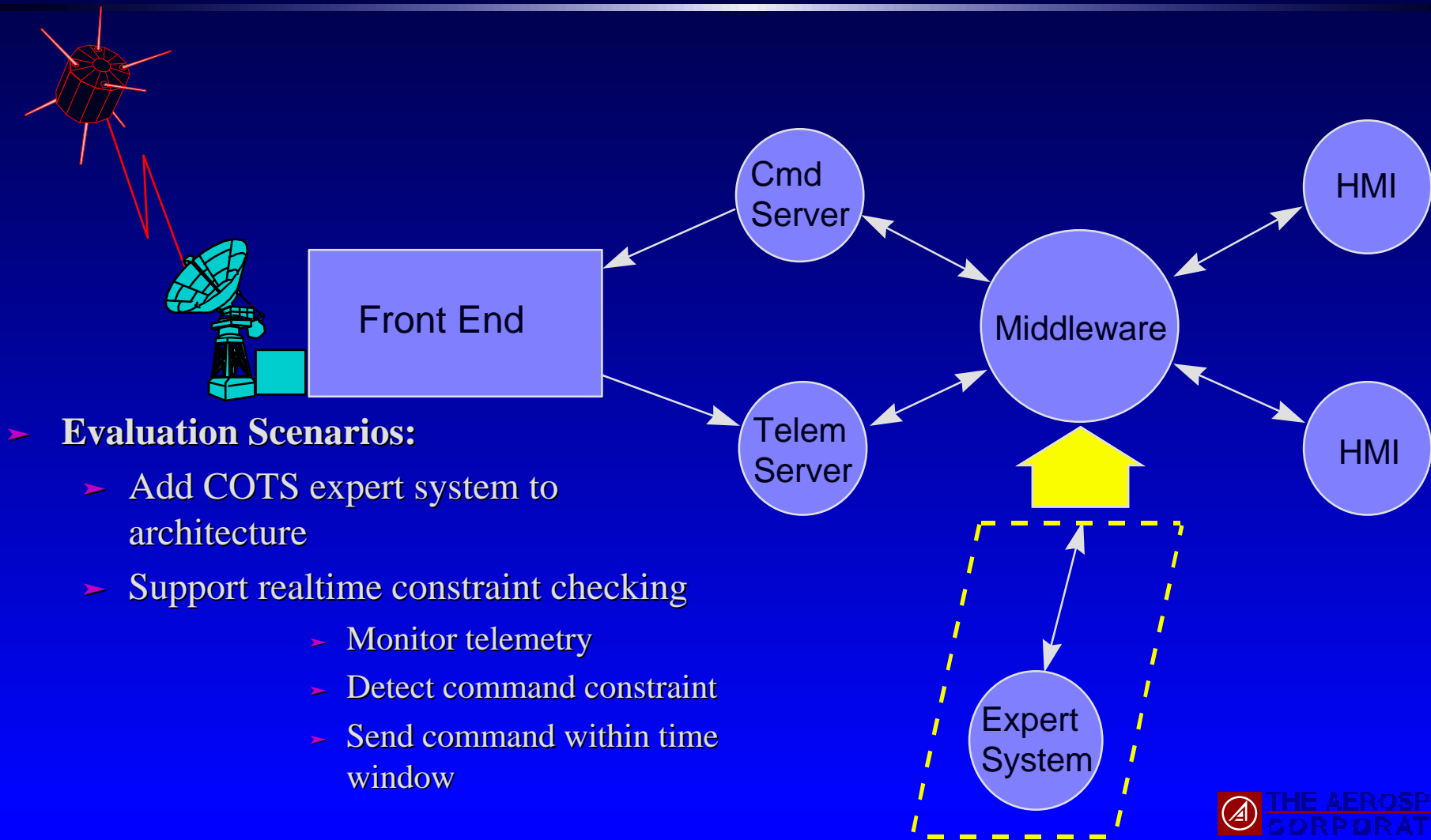
- **Metrics** - *questions focusing on observable measurements of the architecture*
- **Examples for SGS domain:**
  - How many top level components are there in the architecture?
  - What are the number of classes defined for the architecture?
  - How deep are the architecture class hierarchies?
  - What are the fan in / fan out of components?
  - What is the maximum telemetry distribution rate?
  - What is the timing accuracy for commanding?
  - How many clients can the telemetry server handle?

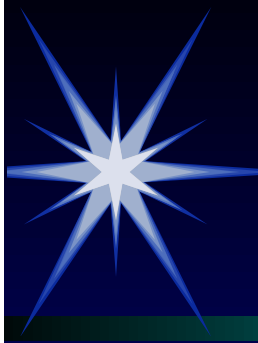


# Methods and Examples

- Simulation, Prototype, Experiment - *evaluation activities designed to help clarify the architecture*
- Example: Model architecture using an executable architecture description language ADL
  - Rapide ADL architecture analysis can clarify issues related to:
    - dynamic behavior and causality
    - component interfaces and behavior
    - scalability of architecture
    - performance requirements
    - changes to requirements

# Example Evaluation





# Summary & Status

- Developing common evaluation criteria for SGS
- Based largely on domain experience base at Aerospace
- Following the recommended evaluation practices defined by SEI
- Continuing to add and rank evaluation criteria, and explore evaluation methodologies
- Final report to be completed by 9/98