Outbrief for Breakout Session 10A

Architecture Methods and Frameworks

James N Martin
Mark W Maier
Reconnaissance Systems Division
Chantilly, Virginia
Major Discussion Points

- What kinds of architectures are we talking about?
  - The two major camps have differing viewpoints
  - What are the key differences in approach and purpose?

- Concerns about the C4ISR Architecture Framework and related methods
  - Understanding what it is
  - Practical lessons learned and common mistakes
  - Relationship to other methods (as used in research, pre-acquisition, and system development)
Two Camps on Architecture

● Camp One

- You are given the requirements, you figure out what the top level design is. The top level design is the architecture.

Motto: “Requirements First”
Two Camps on Architecture

● Camp One
  ■ You are **given the requirements**, you figure out what the top level design is. The top level design is the architecture

    **Motto: “Requirements First”**

● Camp Two
  ■ There are **no requirements** at first, though there are usually vague mission needs or technology opportunities
  ■ Architecting is the process of figuring out the requirements as well as some concept of the solution
  ■ **Requirements are a result** of architecting. (Once you have the requirements, you might move into Camp One.)

    **Motto: “Architecture First”**
Two Camps (cont’d)

- Intermediate State (between Camps 1 & 2)
  - There are **vague requirements**. Some may be called mandatory, but probably none really are.
  
  - You investigate alternative solutions and problems. Then you analyze these alternatives with key stakeholders.
  
  - The net result is a set of “**real**” requirements and a “proof-of-concept” solution, which can move you into Camp One.


Discussions about:

- Achieving integration (internal and external)
- Limiting assumptions of the various methods

Discussions about:

- Pros & Cons of the C4ISR Architecture Framework
- Common mistakes

Integration of Architecture Views
CAF - Pros and Cons, Lessons

- Framework originally conceived as something to address *interoperability* issues
- Multiple experiences in having nearly independent groups developing views separately
- From an engineering perspective, the models are not *analyzable*. This causes a great deal of concern among engineering-oriented groups
- Original purpose was to facilitate evaluations of interoperability by having common documentation standards
- One example from the audience is the desire to use the framework to describe a relatively large system, and then use it to comparatively evaluate some contractor alternatives
Many findings and recommendations
...but skip to the last slide
CAF (Continued)

● The framework is being used to organize overhead reconnaissance enterprise (for the NRO)
  ■ Found that dividing operational views into mission areas is quite useful. Each OV component is repeated for each mission.
  ■ Reality is that a complex architecture description is organized into levels. NRO’s architecture framework standard adds some additional views to deal with NRO concerns.

● Questions came up about how the CAF parts match to each other and bridge to engineering models
  ■ No clear answers, though people have had some pragmatically positive experiences

● Origin was in interoperability, a lot of people’s issues come from application to other sets of concerns
  ■ There is a strong need for more clear examples and education, a lot of confusion for those required to use the CAF
Architecture Description Techniques

- **Does the SV assume point-to-point communications and point-to-point models?**
  - Yes, they do. However, there are pragmatic approaches around it. For example, the SV diagrams can be extended to layered communications, which could include broadcast or other routed methods.

- **Is there an automated method of checking consistency among views?**
  - No
  - Some vendors have developed macros to check for consistency, but the CAF does not give guidance on what consistency is necessary or desirable
Findings/Recommendations

- Most of the problems/mistakes seem to revolve around mismatch of purpose and the framework’s supporting methods and tools
  - Groups applying the C4ISR Framework do not have a consistent understanding of its intended use
    - There have been frequent mis-applications (e.g. using only the essential products as a checklist when the system must go to acquisition)
  - There should be better education and practice in properly applying the architecture process
    - A guidance document on best practices, templates for typical application area, and lessons learned are needed
  - Groups must always understand the purpose of the architecture and the architecture description before building the products
    - Make informed choices about the view products, don’t treat the essential products as a “checklist”
Findings/Recommendations

- The C4ISR Architecture Framework as it stands has no standard mechanism to analyze or enforce consistency among views
  - DOD should evolve the CAF to support better consistency checking between views

- There is a mismatch between research efforts on formal architecture description languages and the needs of decision-makers (as represented in the C4ISR Framework)
  - There should be research on the mappings from the vague representations of the CAF to formal methods which support manipulation and analysis