10TH INTERNATIONAL COMMAND AND CONTROL RESEARCH AND TECHNOLOGY SYMPOSIUM
THE FUTURE OF C2

Development of An Integrated Executable C4I Architecture Model Applying Congruent Processes, Methods, and Tools

TRACK : C4ISR/C2 Architecture

Presentor : Dr. Joongyoon Lee
SE Technology Ltd.,
Republic Of Korea
The Army, Navy, Air Force, Joint Chiefs of Staff of ROK have been developed their own C4I systems independently for their own needs.

- There are **interoperability problems** in C4I, sensor and shooter systems of ROK.
- There are **investment duplication of functions** of C4I systems.

Moreover, there were **incident requirements** during the development of these C4I systems.

- E.g.: The C4I systems shall have counterfire capability until 2005.
Problem solving strategy

We propose an architecture-driven approach as an effective means to:

- enhances the interoperability among the C4I systems of ROK.
- accommodate the incidental requirements quickly.

This paper shows the congruent process, methods and tools which were used for developing an executable architecture model.
J. Armstrong & J. Martin state that the PMTE paradigm is needed to close the gap between ‘what’ should be accomplished and ‘how’ to accomplish the ‘what’.

This study is about the Process, Method, and Tool of the Architecting Process.
Study Object: Congruence

Development of An Integrated Executable C4I Architecture Congruent Processes, Methods, and Tools

- Process
  - Systems Engineering Process
    - MIL-STD-499, EIA 632
- Method
  - DoDFAF
- Tool
  - CORE, Architecture Template
- Environment
  - Training (Process, Method, Tool for IPT)
## Scope & Purpose

<table>
<thead>
<tr>
<th>Class</th>
<th>Contents of scope</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mission</td>
<td>Five missions which have significant real time requirements</td>
</tr>
<tr>
<td>Time Frame</td>
<td>AS-IS(2000-2007) which include currently developing C4I systems</td>
</tr>
<tr>
<td>Organization</td>
<td>Army, Navy, Air Force, and the Joint Chiefs of Staff of ROK</td>
</tr>
<tr>
<td>Geography</td>
<td>Korean territory including land, sea, and airspace.</td>
</tr>
<tr>
<td>System Context</td>
<td>The warfighter components and JCCS-K of USFK are jointed as external systems.</td>
</tr>
</tbody>
</table>
This study introduced the system design process of EIA 632 as an architecting process. And use SE philosophy like that the requirement space and solution space shall be divided according to EIA 632.
Data-centric system design process

An operational concept document can be mapped to the operational view of the architecture.
A system specification can be mapped to system view of the architecture.
The product order takes advantage of the related nature of the products and the dependencies among products.

This chronology does not imply a rigid course of events; however, there is an order of precedence that is required to ensure data integrity.

Dotted circles have different sequences in this paper.
Integrated nature & Order of precedence

OV-1 and OV-4 were used as input information, and then the principal products (OV-5 and OV-6) were developed.

If the principal products were developed first, the other products (OV-2, OV-3, and OV-7) can be developed with little additional effort since the most of operational architecture information is already contained in the principal products.

Because of these dependencies among products, an integrated capability of architecting tool is important from the view point of consistency and efficiency.

The OV-5 was used to capture detailed operational requirements and the SV-4 was developed to satisfy the operational requirements of OV-5.

Developing the principal system architecture products (SV-4 and SV-10) was a major task of system design.

The SV-5 and SV-6 were used to verify that all the operational requirements were implemented in the system architecture.
Spirally Evolving Architectures

Quote: DoD Architecture Framework Ver.1. Deskbook

Spirally Evolving to Integrated/Interoperable Command and Control

Operational View Evolution
System View Spiral
Technical Standards View Evolution

Requirements
CONOPs
Spirally Evolving Architectures

As-Is Architecture

To-Be Architecture

TV-1

SV-8 System evolution forecast
TV-2 Technical standard forecast
SV-9 System technology forecast

Scope of this project
## Product Scope

<table>
<thead>
<tr>
<th>View</th>
<th>Prod. ID</th>
<th>Name</th>
<th>Sel.</th>
</tr>
</thead>
<tbody>
<tr>
<td>AV</td>
<td>AV-1</td>
<td>Overview and Summary Information</td>
<td>O</td>
</tr>
<tr>
<td></td>
<td>AV-2</td>
<td>Integrated Dictionary</td>
<td>O</td>
</tr>
<tr>
<td>OV</td>
<td>OV-1</td>
<td>High-Level Operational Concept Graphic</td>
<td>O</td>
</tr>
<tr>
<td></td>
<td>OV-2</td>
<td>Operational Node Connectivity Description</td>
<td>O</td>
</tr>
<tr>
<td></td>
<td>OV-3</td>
<td>Operational Information Exchange Matrix</td>
<td>O</td>
</tr>
<tr>
<td></td>
<td>OV-4</td>
<td>Organizational Relationships Chart</td>
<td>O</td>
</tr>
<tr>
<td></td>
<td>OV-5</td>
<td>Operational Activity Model</td>
<td>O</td>
</tr>
<tr>
<td></td>
<td>OV-6</td>
<td>Operational Activity Sequence and Timing Descriptions</td>
<td>O</td>
</tr>
<tr>
<td></td>
<td>OV-7</td>
<td>Logical Data Model</td>
<td>O</td>
</tr>
<tr>
<td>SV</td>
<td>SV-1</td>
<td>Systems Interface Description</td>
<td>O</td>
</tr>
<tr>
<td></td>
<td>SV-2</td>
<td>Systems Communications Description</td>
<td>O</td>
</tr>
<tr>
<td></td>
<td>SV-3</td>
<td>Systems-Systems Matrix</td>
<td>O</td>
</tr>
<tr>
<td></td>
<td>SV-4</td>
<td>Systems Functionality Description</td>
<td>O</td>
</tr>
<tr>
<td></td>
<td>SV-5</td>
<td>Operational Activity to Systems Function Traceability Matrix</td>
<td>O</td>
</tr>
<tr>
<td></td>
<td>SV-6</td>
<td>Systems Data Exchange Matrix</td>
<td>O</td>
</tr>
<tr>
<td></td>
<td>SV-7</td>
<td>Systems Performance Parameters Matrix</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>SV-8</td>
<td>Systems Evolution Description</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>SV-9</td>
<td>Systems Technology Forecast</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>SV-10</td>
<td>Systems Functionality Sequence and Timing Descriptions</td>
<td>O</td>
</tr>
<tr>
<td></td>
<td>SV-11</td>
<td>Physical Schema</td>
<td>O</td>
</tr>
<tr>
<td>TV</td>
<td>TV-1</td>
<td>Technical Standards Profile</td>
<td>O</td>
</tr>
<tr>
<td></td>
<td>TV-2</td>
<td>Technical Standards Forecast</td>
<td>X</td>
</tr>
</tbody>
</table>

Exclude To-Be Architecture related products
Tool 1: Templates for elicitation and communication

System Developers

System Users

Communication Barrier

Technical Language

User Language

<table>
<thead>
<tr>
<th>Operational Behavior Data</th>
<th>Operational Information Attribute Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>T1</td>
<td></td>
</tr>
<tr>
<td>T2</td>
<td></td>
</tr>
<tr>
<td>T3</td>
<td></td>
</tr>
<tr>
<td>T4</td>
<td></td>
</tr>
<tr>
<td>T5</td>
<td></td>
</tr>
<tr>
<td>T6</td>
<td></td>
</tr>
<tr>
<td>T7</td>
<td></td>
</tr>
<tr>
<td>T8</td>
<td></td>
</tr>
</tbody>
</table>

ELICITATION & COMMUNICATION Templates
OA Template

An operational architecture template with elicited data

| When | Who | What (I.O) | What (D.O) | How | Verb | Proces. Time (Sec) | Languag e | Trans. Type | Periodicit y | Timeline ss (Sec) | Integrity Check | Secu. Level | Secu. Type | Disemin. Ctrl | LISI |
|------|-----|------------|------------|-----|------|-------------------|-----------|-------------|--------------|----------------|----------------|-------------|-----------|------------|-------------|------|
| T1   |     |            |            |     |      | 60                | K/E       | Direct      | NS           | 60             | 1               | 2           | 6         | OCC       | 2             |
| T2   |     |            |            |     |      | 30                | K/E       | Direct      | NS           | 30             | 1               | 2           | 6         | OCC/FSC   | 3             |
| T3   |     |            |            |     |      | 10                | -         | -           | -            | -              | -              | -           | -         | -         | -             |
| T4   |     |            |            |     |      | 10                | -         | -           | -            | -              | -              | -           | -         | -         | -             |
| T5   |     |            |            |     |      | 10                | K/E       | Direct      | NS           | 10             | 1               | 2           | 6         | ASOC      | 2             |
| T6   |     |            |            |     |      | 10                | K/E       | Direct      | NS           | 10             | 1               | 2           | 6         | X-ATKer   | 2             |
| T7   |     |            |            |     |      | 60                | -         | -           | -            | -              | -              | -           | -         | -         | -             |
| T8   |     |            |            |     |      | 10                | K/E       | Direct      | NS           | 10             | 1               | 2           | 6         | OCC/FSC   | 2             |


4W1H1V Method
Operational behavior description method
Reveal operational behaviors and interact items

Op. Information Attribute Data
Process time, Language, Transaction type, Periodicity, Timeliness, Integrity,
Security level, Security type, Dissemination control, Target LISI
SA Template

A system architecture template filled with system architecture data

<table>
<thead>
<tr>
<th>System Function Data</th>
<th>System Information Attribute Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>When</td>
<td>Who</td>
</tr>
<tr>
<td>T1</td>
<td></td>
</tr>
<tr>
<td>T2</td>
<td></td>
</tr>
<tr>
<td>T3</td>
<td></td>
</tr>
<tr>
<td>T4</td>
<td></td>
</tr>
<tr>
<td>T5</td>
<td></td>
</tr>
<tr>
<td>T6</td>
<td></td>
</tr>
<tr>
<td>T7</td>
<td></td>
</tr>
</tbody>
</table>

System Functions

4W1H1V Method
- System function description method
- Reveal system functions and interact items

System IER Data
- Process time, Language, Transaction type, Periodicity, Timeliness, Integrity,
- Security level, Security type, Dissemination control, Target LISI
We used the CORE® of the Vitech Corp. as a tool.

From DoDAF “Executable architecture” refers to the use of dynamic simulation software to evaluate architecture models.

The CORE® has the capability to build executable architecture models. The following objectives can be achieved using CORE®,

(1) The architecture model itself can be verified for internal self-consistency.
(2) Operational concepts can be simulated, observed dynamically, verified and refined.
(3) Operational plans can be examined and assessed.
(4) Tradeoffs between systems can be assessed.
(5) Architecture measures can be evaluated (given that metrics have been defined), which can support cost-benefit analyses and quantitative acquisition decisions.
An example of architecting template and model
e.g.: Op. activity and Op. IERs (OV-3)
Operational Activity Model

Interface informations between operational activities
BPR

• Operational Effectiveness Analysis

Readyness: XX min.  X min.

Time to Readyness
Operational Node Connectivity Description (OV-2)
Architecture Elements Alignment


Alignment

Operational Model

System Model

Op. Activities vs. System Functions (SV-5)
Architecting Process

1. Stakeholders
   Concepts
   OPLAN

2. Gather & Refine using OA Template

3. Modeling and Simulation using CORE®

4. OV Product Output

5. Gather & Refine using SA Template

6. Modeling and Simulation using CORE®

7. SV Product Output
Architecting Process Using Templates

A. Templates can be used as communication tool
   • Between stakeholders and system engineers.
   • Between system engineers and CaSysE Tool.

A. Templates can decrease the rework very effectively.

<table>
<thead>
<tr>
<th>Stakeholders</th>
<th>A. Templates</th>
<th>System Engineer</th>
<th>A. Templates</th>
<th>CaSysE Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Summary

- Defined sequence
- Sys. Architecture Template Using Methods

Supported by

**Process**

**Method**

**Tools**

CaSysE Tool

Architecting Templates

Requirements Definition Process

Solution Definition Process
Results

This paper introduced congruently integrated process, methods, and tools for architecture model development.
- The system design process of EIA 632,
- The DoDAF was used as a method to implement the process.
- As an architecting tool, the architecture templates and specialized systems engineering tool CORE® were used.

The operational architecture template was effective
- in eliciting requirements,
- to reduce requirement fluctuation, and
- to adopt incident requirements quickly.

Due to the integrated architecture capability of CORE®,
- the architecture development effort was reduced.

Due to the executable architecture capability of CORE®,
- the performance of an architecture was enhanced.

The congruently integrated process, method, and tools were effective and efficient
- in solving interoperability problems by identifying IERs and
- in delivering key interface requirements and key functional performance requirements to each system of warfighter domain.

Finally, the interoperability level 3 (LISI 3) of C4ISR systems was assured from the architecture model demonstration.