The Use of Patterns in Systems Engineering

Satya Moorthy

Robert Cloutier, Ph.D.

Lockheed Martin – MS2

Topics

- ***Abstract***
- ***Definitions***
- ***Value of Patterns***
- ***Documented Pattern Language***
- ***Patterns***
- ***New Pattern Language***
- ***Summary***
- ***References***

About the Authors

***Satya Moorthy** is currently a Systems Engineer with Lockheed Martin in Moorestown NJ. He develops architectures for complex systems using systems engineering tools and architecture frameworks*

Satya has as a M.E. in Systems Engineering, and a B.E. in Computer Engineering, both from Stevens Institute of Technology. He is also a Certified Enterprise Architect from the Federal Enterprise Architecture Certification (FEAC) Institute

About the Authors

***Robert Cloutier** received his Ph.D. in Systems Engineering from Stevens Institute of Technology. He currently is a Principal Systems Engineer with the Lockheed Martin Corporation in Moorestown, NJ. He is responsible for developing and modeling architectures for complex systems, and has over 20 years experience in systems engineering, software engineering, and project management in both commercial and defense industries.*

Rob also has an M.B.A. from Eastern College, and a B.S. from the United States Naval Academy. He is an Industry Fellow at Stevens Institute of Technology and an Adjunct Professor for Eastern University. He is a member of the International Council on Systems Engineering (INCOSE), an associate editor for the Journal of Enterprise Architecture, and is a member of IEEE. Rob also chairs the Rowan University Electrical and Computer Engineering Department Industry Advisory Board. Finally, Rob teaches Architecture and Design for Stevens Institute's Systems Engineering Graduate program.

Abstract

- ***A pattern is a solution that can be applied to similar problems, or problems with similar characteristics. Patterns have existed since before most people can remember, and have been used across different disciplines from designing a building, to developing software, to making clothing.***
- ***Recent research has examined the use of patterns in systems engineering and at the documentation of such patterns to enable reuse. Based on a documented systems engineering pattern language from this recent research, this presentation reviews the documented pattern language, identifies the patterns used to develop the pattern language, and then develops a new pattern language based on an existing functional sequence.***

Architect's Challenge

- ***The art-like quality of systems architecting depends on the architect's ability to recognize complex system requirements patterns and the ability to match those patterns to architecture solutions [Carpenter, 1996]***
- ***Through years of experience, the architect recognizes relationships and patterns***
  - ***Applies correct solution to the problem at hand***

What Constitutes a Pattern?

Some patterns seem self evident, but have actually matured out of experience…

Simple at first…

They are the result of trial and error, but survive the test of time.

And maybe they become more complex

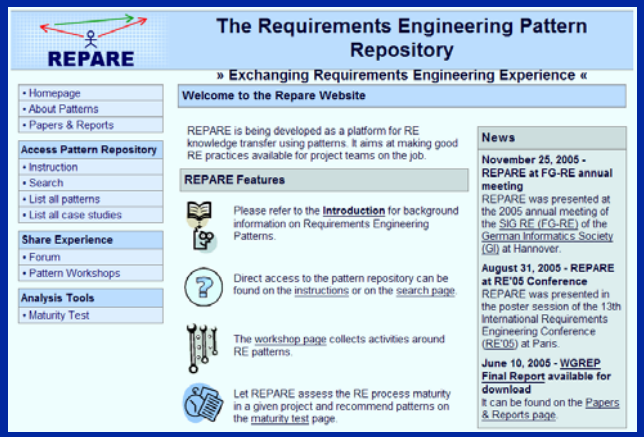
Patterns are Reusable Successes

Definitions

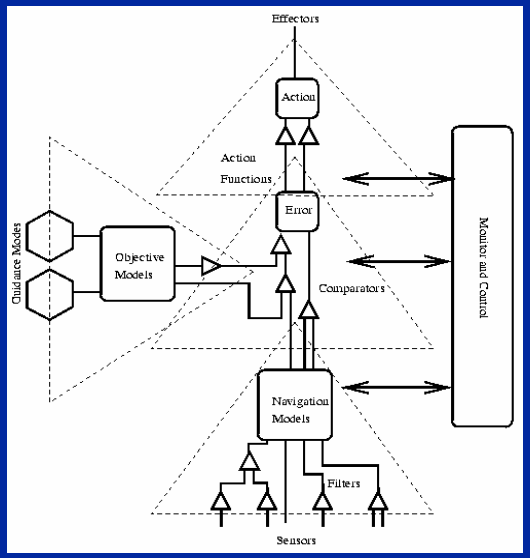
- ***System architecture patterns constitute high-level structures, appropriate to the design of the major components of a system. They express the relation between the context, a problem, and a solution, documenting attributes and usage guidance. They are time-proven in solving problems similar in nature to the problem under consideration. [Cloutier, Doctoral Dissertation, 2006]***

Patterns are not invented, patterns are mined from past successful designs

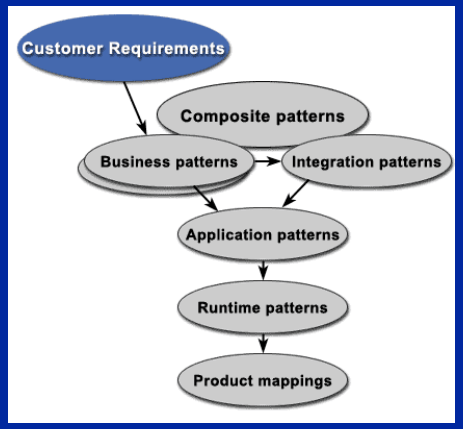
Patterns In Use by Multiple Disciplines



Requirements Engineering



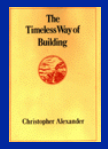
Control Systems Engineering
http://g.oswego.edu/dl/acs/acs/acs.html



IBM e-Business Pattern Hierarchy

Pattern History

- *1964 – Christopher Alexander*
  - *Books on Architecture, building and urban planning*
  - *Notes on the Synthesis of Form*
  - *A Pattern Language*
  - *A Timeless Way of Building*

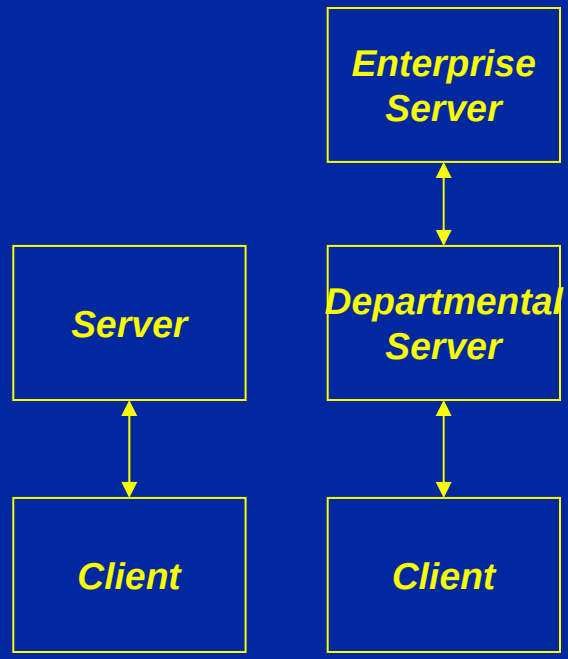


- *1987 - Ward Cunningham & Kent Beck*
  - *Decided to use some of Alexander's ideas*
  - *Developed five patterns for guiding novice Smalltalk programmers*
  - *Presented paper at OOPSLA'87 in Orlando*
    - **"Using Pattern Languages for Object-Oriented Programs".**
- *1995 - Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides*
  - *Software Design*
  - *Design Patterns: Elements of Reusable Object-Oriented Software*

What is a Pattern?

A pattern is a model or facsimile of an actual thing or action, which provides some degree of representation (an abstraction) to enable the recreation of that entity over and over again.

"Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem in such a way that you could use this solution a million times over without doing it the same way twice." [Ale77].



Value of Patterns

- *Knowledge Management*
  - *"… mining the patterns of classic embedded systems to capture the core competencies of their business… Why? We can trace availability and fault tolerance to patterns, and we have extracted those patterns from the minds of long-standing experts." [Cop97]*
- *Capture Good Architecture Concepts*
  - *Enable reuse of good concepts and implementations, and to preserve them for future projects*
- *Control Complexity*
  - *Architectural patterns may help control the complexity of an architecture by standardizing it on a well known and practiced pattern*
- *Common Understanding*
  - *Describe parts of the designs and implementations in the context of known and understood patterns may foster a common understanding of the architecture*
- *Mitigate Risks*

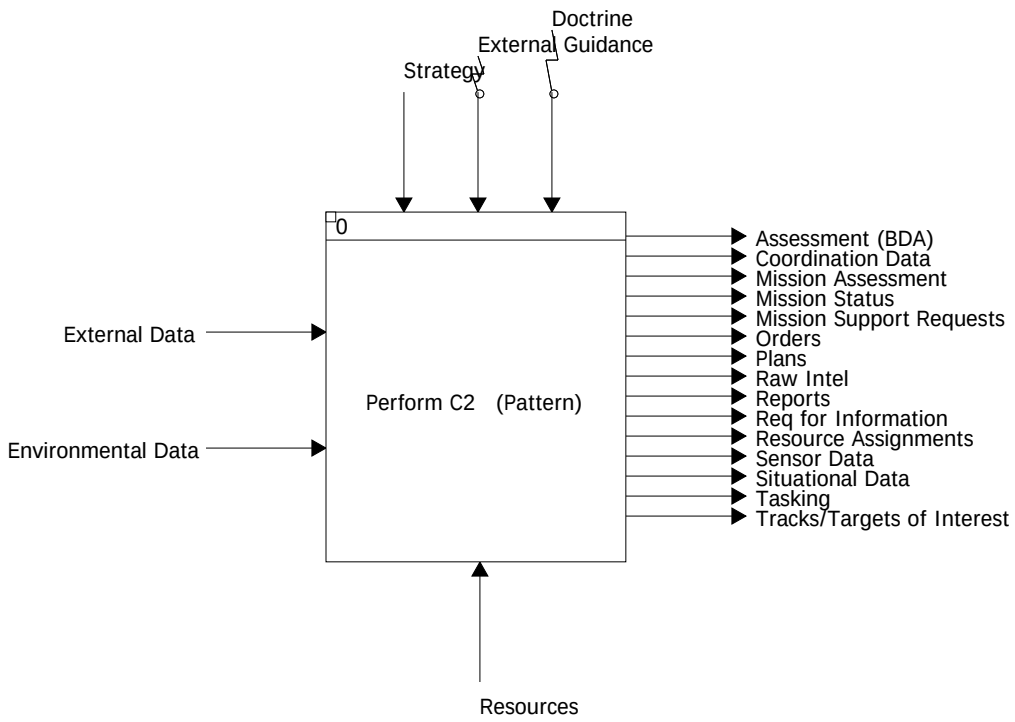
Documented Pattern

Pattern Name:	***Perform C2***
Aliases:	***None known***
Keywords:	***Plan Detect Control Act C2***
	Command Control
Problem Context:	***Does not address "Prepare" precondition nor "Assess" post condition***
Problem Description:	***In command and control (C2), it is normal for the problem to progress from one stage to another. Those stages are Plan/Detect/Control/Act***
Forces:	***Terminology from various domains may be different, and must be adapted in the application of the pattern***

Perform C2 Pattern

Pattern Solution:	***This pattern provides the basis for developing the command and control (C2) interfaces and information that moves through the stages of C2. It provides the A0 Context and the first level of decomposition using IDEF0.***
Interfaces:	***Information flows between the stages of this pattern, as well as feedback loops. Some information is generated only in a particular stage and then output in the form of reports. Names of information can be modified as required by specific domain application.***

Perform C2 Pattern

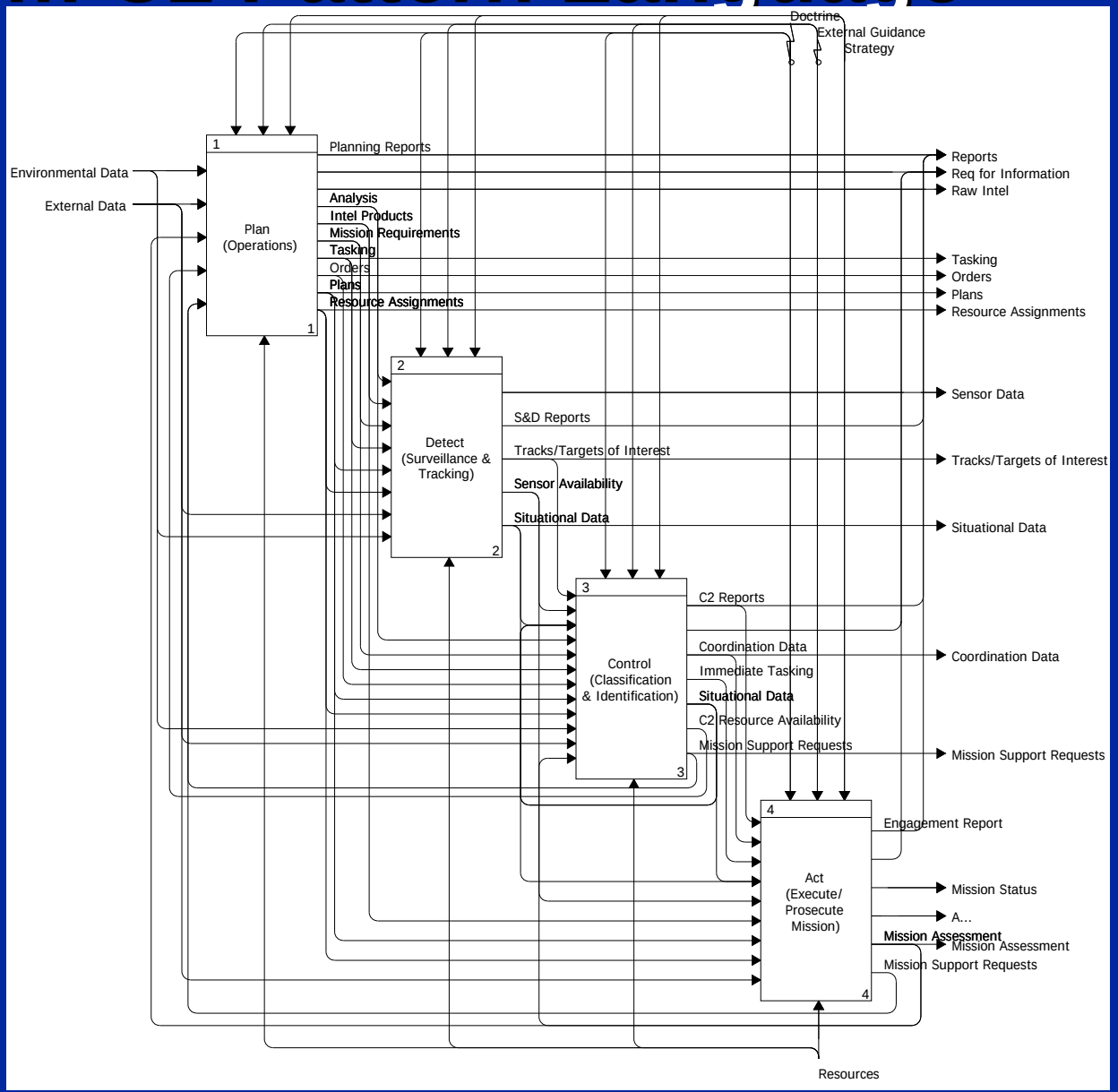


Date: Tuesday, January 17, 2006	Author: Robert J. Cloutier
Number: 0	Name: Perform C2 (Pattern)

Pattern Languages

- ***A pattern language is a network of patterns that are complementary, and may work together to form a larger pattern. The C2 pattern is actually a pattern language. It is comprised of four architecture smaller patterns – plan, detect, control and act. Each of these four patterns could be used independently to architect the Concept of Operations (CONOPS) of another system. For instance, the Plan pattern shown could be used by a marketing organization developing a new software application to manage new product launches.***

Perform C2 Pattern Language



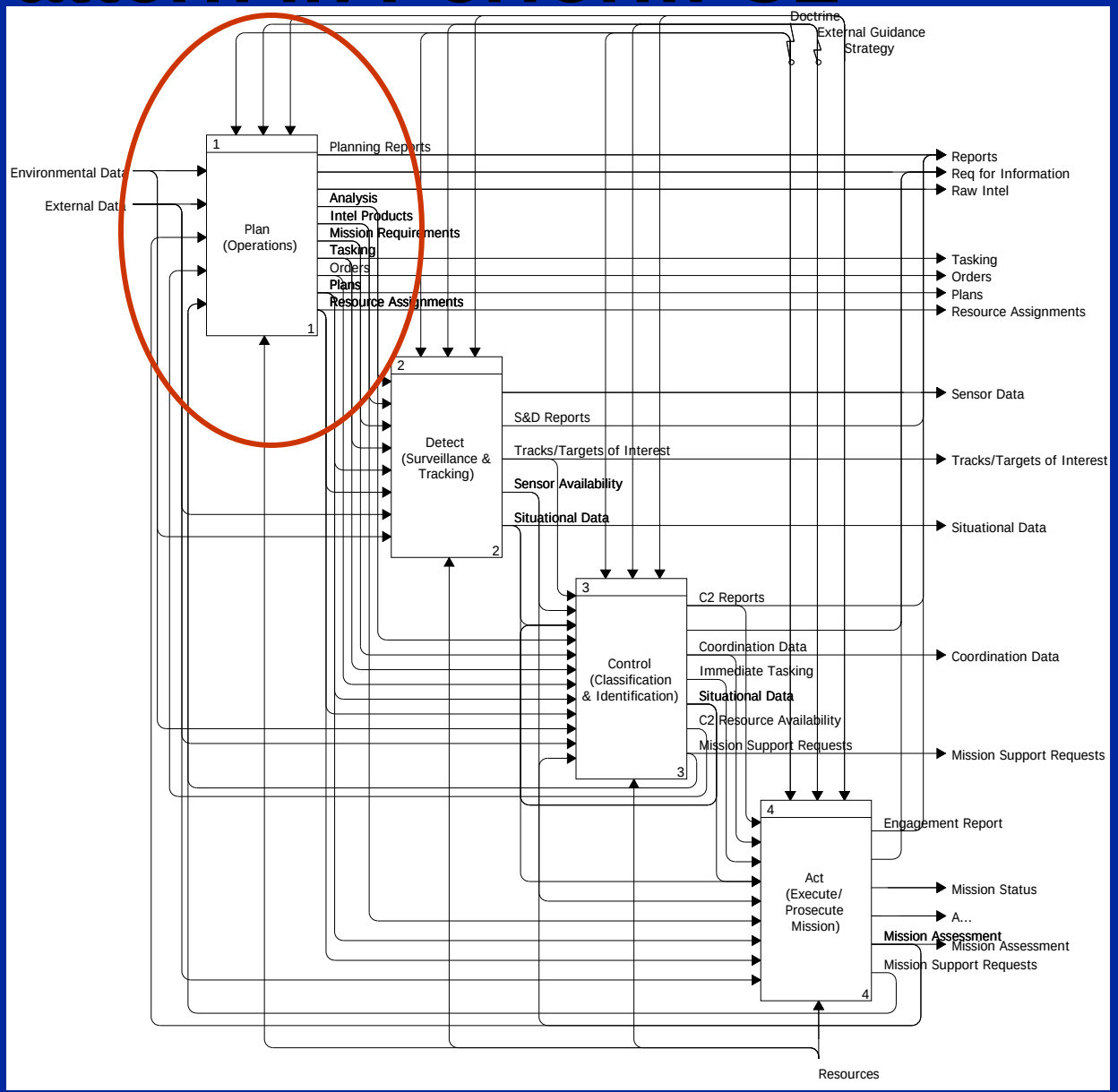
Perform C2 Pattern Language

- *The Perform C2 Pattern Language is comprised of four patterns*
  - *Plan*
  - *Detect*
  - *Control*
  - *Engage*
- *The patterns can be reused by an architect who is starting to develop an activity model based on another functional sequence*

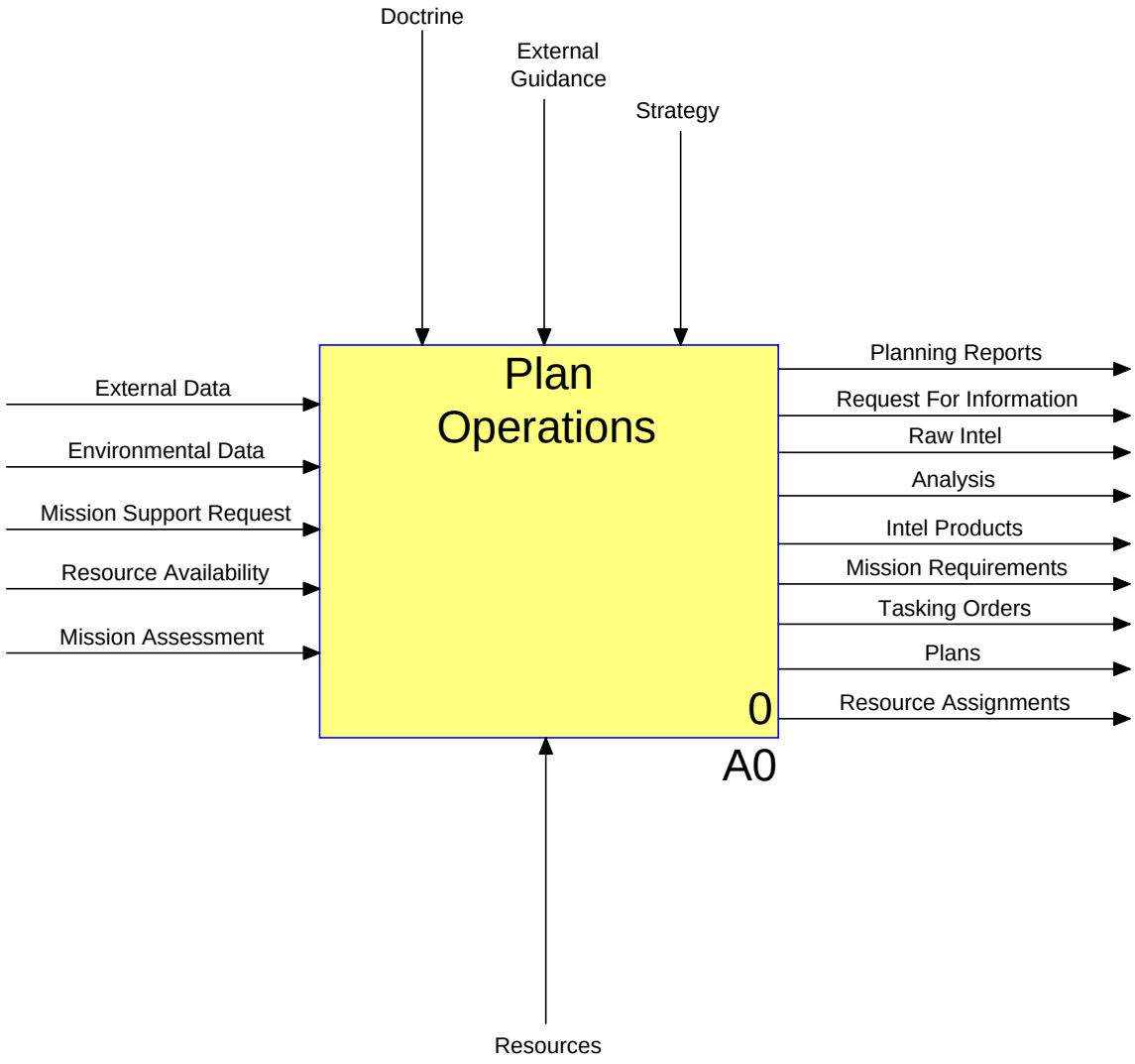
New Pattern Language

- *Reuse of elements of Perform C2 Pattern to develop a new Pattern Language based on another functional sequence.*
- *Example – MAPE*
  - *Monitor the Situation*
  - *Assess the Situation*
  - *Plan Operations*
  - *Execute the Mission*
- *Reuse "Plan" from previously documented pattern in order to develop new pattern language based on MAPE*

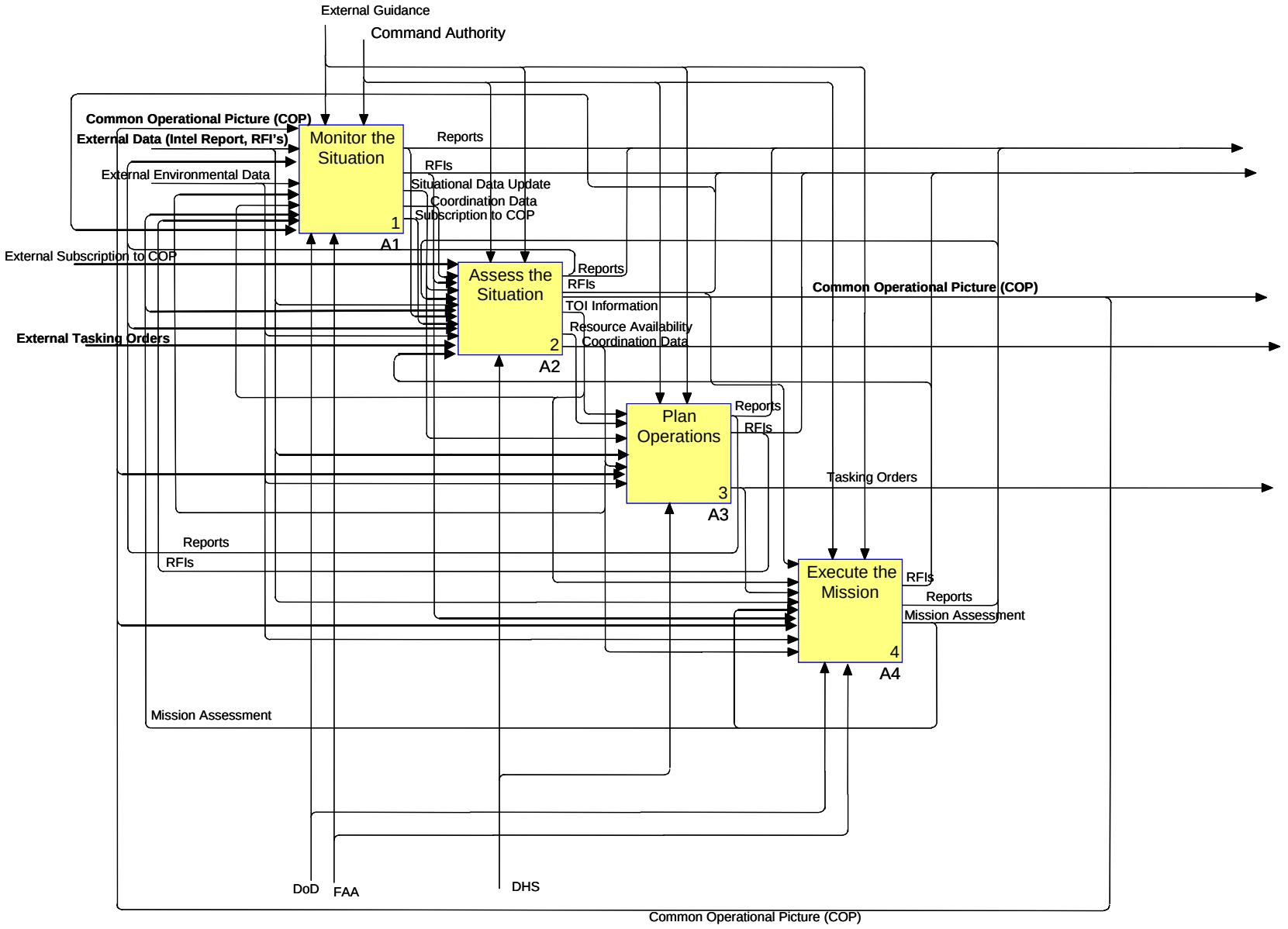
Plan Pattern in Perform C2



“Plan” Pattern



Reusing “Plan” in MAPE - Situational Awareness



Summary

- ***The use of patterns aids systems engineers in solving similar problems***
- ***Patterns can be combined into pattern languages***
- ***Documenting patterns can be useful for architecting systems in the future***
- ***Pattern documentation is not confined to, or limited by, engineering tools***

References

Alexander, Christopher, Notes on the Synthesis of Form, Cambridge: Harvard University Press, 1964.

Alexander, Christopher, A Pattern Language, New York: Oxford University Press, 1977.

Alexander, Christopher, A Timeless Way of Building. New York: Oxford University Press, 1979.

Alexander, Christopher, The Order of Nature: An Essay on the Art of Building and the Nature of the Universe, Book 1 The Phenomenon of Life, Berkley, CA: The Center for Environmental Structure, 2002.

Carpenter, Robert Glenn. System Architect's Job Characteristics and Approach to the Conceptualization of Complex Systems. **Doctoral Dissertation** presented to the Faculty of the Graduate School University of Southern California, May 1996.

Cloutier, Robert and Dinesh Verma, Applying Pattern Concepts to Enterprise Architecture, Journal of Enterprise Architecture Volume 2, Number 2, May 2006.

Cloutier, Robert, Applicability of Patterns to Architecting Complex Systems, Doctoral Dissertation, Stevens Institute of Technology, 2006.

Cloutier, Robert, Towards the Application of Patterns to Systems Engineering, Proceedings of the 2005 Conference on Systems Engineering Research, March 2005.

Cloutier and Verma, Applying the Concept of Patterns to Systems Architecture, INCOSE Systems Journal, Pre-release.

Cloutier, Robert, Calimar Consulting Website, http://www.calimar.com/research.htm

References

Coplien, James, Idioms and Patterns as Architectural Literature, IEEE Software Special Issue on Objects, Patterns, and Architectures, January 1997.

Hartnett, Kenneth , Enhanced Civilian/Military Airspace Security (ECMAS) System Conceptual Architecture, Spring 2005

Contact Information

Satya Moorthy
Lockheed Martin MS2, Moorestown NJ
E-mail : satya.n.moorthy@lmco.com

Robert Cloutier, Ph.D.
Lockheed Martin MS2, Moorestown NJ
E-mail : robert.j.cloutier@lmco.com
robert.cloutier@stevens.edu

Any Questions?